(19)日本国特許庁（ＪＰ）　(12)**公表特許公報（Ａ）**　(11)特許出願公表番号
特表2002－514813
（P2002－514813A）
(43)公表日　平成14年５月21日(2002.5.21)

(51)Int.Cl.[7]	識別記号	ＦＩ		テーマコード（参考）
Ｇ０６Ｆ　13/00	３０１	Ｇ０６Ｆ　13/00	３０１Ｑ	５Ｂ０１４
11/14	３１０	11/14	３１０Ｆ	５Ｂ０２７
11/20	３１０	11/20	３１０Ａ	５Ｂ０３４
12/00	５１４	12/00	５１４Ｅ	５Ｂ０８２
	５４５		５４５Ａ	５Ｂ０８３

審査請求　未請求　予備審査請求　有　（全 70 頁）　最終頁に続く

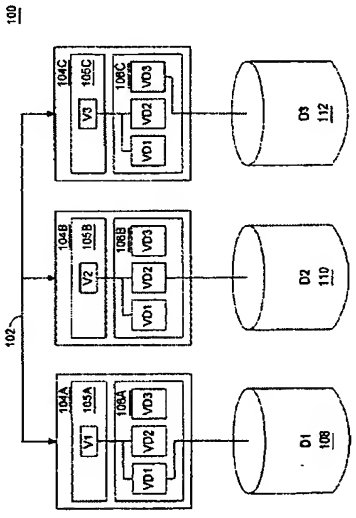
(21)出願番号　特願2000－548806(P2000－548806)
(86)（22)出願日　平成11年５月７日(1999.5.7)
(85)翻訳文提出日　平成12年11月10日(2000.11.10)
(86)国際出願番号　ＰＣＴ／ＵＳ９９／０９９０３
(87)国際公開番号　ＷＯ９９／５９０６４
(87)国際公開日　平成11年11月18日(1999.11.18)
(31)優先権主張番号　０９／０７６，３８８
(32)優先日　平成10年５月12日(1998.5.12)
(33)優先権主張国　米国（ＵＳ）
(31)優先権主張番号　０９／０７６，３４７
(32)優先日　平成10年５月12日(1998.5.12)
(33)優先権主張国　米国（ＵＳ）

(71)出願人　サン・マイクロシステムズ・インコーポレーテッド
SUN MICROSYSTEMS, INCORPORATED
アメリカ合衆国　94303　カリフォルニア州・パロ　アルト・サン　アントニオ　ロード・901
(72)発明者　スローター，グレゴリー・エル
アメリカ合衆国・94306・カリフォルニア州・パロ　アルト・エマーソン　ストリート・3326
(74)代理人　弁理士　山川　政樹

最終頁に続く

(54)【発明の名称】　高可用性クラスタ仮想ディスク・システム

(57)【要約】
クラスタは、そのクラスタの各記憶装置へのアクセス権をそのクラスタの各ノードに与える仮想ディスク・システムを実現する。この仮想ディスク・システムは、障害が存在する状態で記憶装置にアクセスすることができ、データ・アクセス要求が確実に完了するような高い可用性を有する。ノード間で一貫したマッピングおよびファイル許可データを保証するために、データは高可用性クラスタ・データベースで記憶される。クラスタ・データベースは障害が存在する状態でもノードに一貫したデータを提供するので、各ノードは一貫したマッピングおよびファイル許可データを有することになる。ノード間のリンクを確立し、そのリンクを管理するクラスタ・トランスポート・インタフェースが提供される。クラスタ・トランスポート・インタフェースが受け取るメッセージは１つまたは複数のリンクを介して宛先ノードに運搬される。クラスタの構成は動作中に変更することができる。構成を変更する前に、再構成手順はデータ・アクセス要求を中断し、保留中のデータ・アクセス要求が完了するのを待つ。新しい構成を反映するために、再構成が



【特許請求の範囲】

【請求項１】　第１のノードと、第２のノードと、前記第１のノードと前記第２のノードとの間に結合された通信リンクと、

前記第１のノードに結合された記憶装置であって、前記記憶装置が前記第２のノードに結合されているように見えるように構成された仮想ディスク・システムを使用して前記記憶装置にアクセスするように前記第２のノードが構成される記憶装置とを含み、

前記仮想ディスク・システムが前記第２のノード上で動作するドライバと前記第１のノード上で動作するマスタとを含み、前記第２のノードが前記仮想ディスク・システムの仮想ディスクにアクセスすると、前記ドライバは前記マスタにデータ要求を送るように構成され、前記マスタは前記記憶装置からのデータにアクセスするように構成され、前記マスタは前記通信リンクを介して前記ドライバに応答を送るように構成され、

前記ドライバは前記ドライバが前記応答を受け取るまで前記データ要求のコピーを記憶するように構成され、前記ドライバは前記ドライバが前記応答を受け取り損なった場合に前記要求を再送するように構成される、分散コンピューティング・システム。

【請求項２】　前記データ通信インタフェースおよび前記記憶装置に結合された第３のノードをさらに含み、前記ドライバは前記ドライバが前記応答を受け取り損なった場合に前記第３のノード上の第２のマスタに前記データ要求を再送するように構成され、前記第２のマスタは前記記憶装置に関するデータにアクセスするように構成される、請求項１に記載の分散コンピューティング・システム。

【請求項３】　前記ドライバが前記第１のノードまたは通信リンクの障害のために前記データを受け取り損なった場合、前記ドライバは前記分散コンピューティング・システムが再構成したあとで前記データ要求を再送するように構成される、請求項１に記載の分散コンピューティング・システム。

【請求項４】　前記再送データ要求が、前記記憶装置に結合された第３のノードに送られる、請求項３に記載の分散コンピューティング・システム。

【請求項５】　前記仮想ディスク・システムが前記記憶装置と通信するために一次ノードと代替ノードとを維持し、前記第２のノードは前記第２のノードが応答を受け取り損なった場合に前記一次ノードに前記データ要求を送り、前記代替ノードに前記データ要求を再送するように構成される、請求項４に記載の分散コンピューティング・システム。

【請求項６】　前記第１のノードが前記一次ノードであり、前記第３のノードが前記代替ノードである、請求項５に記載の分散コンピューティング・システム。

【請求項７】　第１のノードと、第２のノードと、前記第１のノードと前記第２のノードとの間に結合された通信リンクと、

前記第１のノードに結合された記憶装置であって、前記記憶装置が前記第２のノードに結合されているように見えるように構成された仮想ディスク・システムを使用して前記記憶装置にアクセスするように前記第２のノードが構成される記憶装置とを含み、

前記仮想ディスク・システムは前記記憶装置に対する仮想ディスクのマッピングを含むように構成され、前記第１のノードと前記第２のノードは一貫したマッピング・データを受け取るように構成される、分散コンピューティング・システム。

【請求項８】　前記マッピングが、前記記憶装置に結合されたノードと、前記記憶装置に対応する前記ノード上のディスク装置とを識別し、前記マッピングが前記第１のノードおよび前記第２のノードによってアクセスされる高可用性データベースに記憶される、請求項７に記載の分散コンピューティング・システム。

【請求項９】　前記高可用性データベースがクラスタ構成データベースである、請求項８に記載の分散コンピューティング・システム。

【請求項１０】　前記マッピングが、前記記憶装置に結合された一次ノードと前記記憶装置に結合された代替ノードとを示すデータを含む、請求項７に記載の分散コンピューティング・システム。

【請求項１１】　データ・アクセス要求は前記マッピングが更新されたとき

に中断されるように構成される、請求項７に記載の分散コンピューティング・システム。

【請求項１２】　前記マッピングは、ノードがクラスタに加わるかまたはクラスタを離れるときに更新されるように構成される、請求項１１に記載の分散コンピューティング・システム。

【請求項１３】　データ・アクセス要求が再開されたときにノードが新しいマッピングを求めてデータベースに照会する、請求項１２に記載の分散コンピューティング・システム。

【請求項１４】　動作中に分散コンピューティング・システムの構成を更新することができる、請求項１３に記載の分散コンピューティング・システム。

【請求項１５】　第１のノードと、第２のノードと、前記第１のノードと前記第２のノードとの間に結合された通信リンクと、

前記第１のノードに結合された記憶装置であって、前記記憶装置が前記第２のノードに結合されているように見えるように構成された仮想ディスク・システムを使用して前記記憶装置にアクセスするように前記第２のノードが構成される記憶装置とを含み、

前記記憶装置の許可データが前記第１のノードと前記第２のノードとの間で一貫している、分散コンピューティング・システム。

【請求項１６】　前記許可データは、前記第１のノードおよび前記第２のノードによってアクセスされる高可用性データベースに記憶されるように構成される、請求項１５に記載の分散コンピューティング・システム。

【請求項１７】　前記高可用性データベースがクラスタ構成データベースである、請求項１６に記載の分散コンピューティング・システム。

【請求項１８】　第１のノードと、第２のノードと、前記第１のノードと前記第２のノードとの間に結合された通信リンクと、

前記第１のノードに結合された第１の記憶装置であって、前記第１の記憶装置が前記第２のノードに結合されているように見えるように構成された仮想ディスク・システムを使用して前記第１の記憶装置にアクセスするように前記第２のノードが構成される第１の記憶装置と、

前記第２のノードに結合された第２の記憶装置であって、前記第２の記憶装置が前記第１のノードに結合されているように見えるように構成された仮想ディスク・システムを使用して前記第２の記憶装置にアクセスするように前記第１のノードが構成される第２の記憶装置と、

前記仮想ディスク・システムの上または前記仮想ディスク・システムの下に層状に重ねたボリューム・マネージャとを含む、分散コンピューティング・システム。

【請求項１９】　前記ボリューム・マネージャが前記仮想ディスク・システムの下に層状に重ねられ、前記仮想ディスク・システムはボリュームにアクセスするように構成され、前記ボリュームは前記第１の記憶装置または前記第２の記憶装置にアクセスするように構成される、請求項１８に記載の分散コンピューティング・システム。

【請求項２０】　前記ボリュームが前記仮想ディスク・システムにとって記憶システムとして現れる、請求項１９に記載の分散コンピューティング・システム。

【請求項２１】　前記ボリュームが高可用性記憶装置である、請求項２０に記載の分散コンピューティング・システム。

【請求項２２】　前記ボリューム・マネージャが前記仮想ディスク・システムの上に層状に重ねられ、クライアントが前記仮想ディスク・システムの仮想ディスクにアクセスするように構成されたボリュームにアクセスすることができる、請求項１８に記載の分散コンピューティング・システム。

【請求項２３】　前記ボリュームは２つまたはそれ以上の仮想ディスクにアクセスするように構成される、請求項２２に記載の分散コンピューティング・システム。

【請求項２４】　第１のノードと、第２のノードと、前記第１のノードと前記第２のノードとの間に結合された通信リンクと、

前記第１のノードに結合された記憶装置であって、前記記憶装置が前記第２のノードに結合されているように見えるように構成された仮想ディスク・システムを使用して前記記憶装置にアクセスするように前記第２のノードが構成される記

憶装置とを含み、

前記仮想ディスク・システムが前記第２のノード上で動作するドライバと前記第１のノード上で動作するマスタとを含み、前記第２のノードが前記記憶装置に対応する仮想ディスクにアクセスすると、前記ドライバは前記マスタにデータ要求を送るように構成され、前記マスタは前記記憶装置からのデータにアクセスするように構成され、前記マスタは前記通信インタフェースを介して前記ドライバに前記データを送るように構成され、

前記ドライバは前記ドライバが前記データを受け取るまで前記データ要求のコピーを記憶するように構成され、前記ドライバは前記ドライバが前記データを受け取り損なった場合に前記データ要求を再送するように構成され、前記仮想ディスク・システムは前記記憶装置に対する仮想ディスクのマッピングを含むように構成され、前記第１のノードと前記第２のノードはノード障害が発生した場合に一貫したマッピング・データを受け取るように構成され、前記記憶装置の許可データが前記第１のノードと前記第２のノードとの間で一貫している、分散コンピューティング・システム。

【請求項２５】　前記データ通信インタフェースおよび前記記憶装置に結合された第３のノードをさらに含み、前記ドライバは前記ドライバが前記データを受け取り損なった場合に前記第３のノードに前記データ要求を再送するように構成される、請求項２４に記載の分散コンピューティング・システム。

【請求項２６】　前記マッピングが、前記記憶装置に結合された一次ノードと前記記憶装置に結合された二次ノードとを示すデータを含む、請求項２５に記載の分散コンピューティング・システム。

【請求項２７】　前記マッピングおよび前記許可データは、前記第１のノードおよび前記第２のノードによってアクセス可能な高可用性データベースに記憶されるように構成される、請求項２６に記載の分散コンピューティング・システム。

【請求項２８】　第１のノードと、第２のノードと、前記第１のノードおよび前記第２のノードに結合された通信リンクとを含み、前記第１のノードおよび前記第２のノードが前記記憶装置にアクセスし、

前記記憶装置が前記記憶装置に関連する許可データを有し、前記許可データが前記第１のノードおよび前記第２のノードによってアクセス可能な高可用性分散データベースに記憶され、特定のノードが前記記憶装置をオープンすると、前記特定のノードが前記記憶装置に関する前記許可データによって装置ファイルを作成し、それにより、ノード障害が存在する状態で前記第１のノードと前記第２のノードが一貫した許可データを入手する、分散コンピューティング・システム。

【請求項２９】　前記許可データが、所有者と、グループと、前記所有者および前記グループのための許可モードとを含む、請求項２８に記載の分散コンピューティング・システム。

【請求項３０】　前記許可モードが、読取りと、書込みと、実行とを含む、請求項２９に記載の分散コンピューティング・システム。

【請求項３１】　前記高可用性データベースがクラスタ構成データベースである、請求項２８に記載の分散コンピューティング・システム。

【請求項３２】　前記記憶装置がディスク装置である、請求項２８に記載の分散コンピューティング・システム。

【請求項３３】　前記第１のノードが前記記憶装置に直接アクセスし、前記第２のノードが前記通信リンクを介して前記記憶装置にアクセスする、請求項２８に記載の分散コンピューティング・システム。

【請求項３４】　前記記憶装置が特定のノードによって最初にオープンされたときに前記装置ファイルが作成される、請求項２８に記載の分散コンピューティング・システム。

【請求項３５】　記憶装置を含む分散コンピューティング・システム内の複数のノード間で一貫した許可データを維持する方法であって、

高可用性分散データベースに前記許可データを記憶するステップと、

前記複数のノードのうちの第１のノードが装置をオープンし、前記高可用性データベースにアクセスして前記装置に関する許可データを入手するステップと、

前記複数のノードのうちの第２のノードが前記装置をオープンし、前記高可用性データベースにアクセスして前記装置に関する前記許可データを入手するステップとを含み、

それにより、障害が存在する状態で前記第１のノードと前記第２のノードが一貫した許可データを入手する方法。

【請求項３６】　前記装置をオープンする前記ノードが前記記憶装置に関する装置ファイルを作成し、前記装置ファイルが前記許可データを含む、請求項３５に記載の方法。

【請求項３７】　前記許可データが、所有者と、グループと、前記所有者および前記グループのための許可モードとを含む、請求項３６に記載の方法。

【請求項３８】　前記許可モードが、読取りと、書込みと、実行とを含む、請求項３７に記載の方法。

【請求項３９】　前記高可用性データベースがクラスタ構成データベースである、請求項３５に記載の方法。

【請求項４０】　前記記憶装置がディスク装置である、請求項３５に記載の方法。

【請求項４１】　前記第１のノードが前記記憶装置に直接アクセスし、前記第２のノードが前記データ通信リンクを介して前記記憶装置にアクセスする、請求項３５に記載の方法。

【請求項４２】　前記記憶装置が特定のノードによって最初にオープンされたときに前記装置ファイルが作成される、請求項３６に記載の方法。

【請求項４３】　分散コンピューティング・システム内の複数のノード間で一貫した許可データを維持するためのプログラム命令を含むコンピュータ可読記憶媒体であって、前記プログラム命令が前記分散コンピューティング・システムの前記複数のノード上で実行され、前記プログラム命令が、

高可用性コヒーレント・データベースに前記許可データを記憶するステップと、

前記複数のノードのうちの第１のノードが装置をオープンし、前記高可用性データベースにアクセスして前記装置に関する許可データを入手するステップと、

前記複数のノードのうちの第２のノードが前記装置をオープンし、前記高可用性データベースにアクセスして前記装置に関する前記許可データを入手するステップとを実施するように動作可能であり、

それにより、障害が存在する状態で前記第１のノードと前記第２のノードが一貫した許可データを入手する、コンピュータ可読記憶媒体。

【請求項４４】　前記ファイルが前記記憶装置に関するものであり、前記装置ファイルが前記許可データを含む、請求項４３に記載の媒体。

【請求項４５】　前記許可データが、所有者と、グループと、前記所有者および前記グループのための許可モードとを含む、請求項４４に記載の媒体。

【請求項４６】　前記許可モードが、読取りと、書込みと、実行とを含む、請求項４５に記載の媒体。

【請求項４７】　前記高可用性データベースがクラスタ構成データベースである、請求項４３に記載の媒体。

【請求項４８】　前記記憶装置が特定のノードによって最初にオープンされたときに前記装置ファイルが作成される、請求項４４に記載の媒体。

【請求項４９】　１つの通信リンクに結合された１つまたは複数のノードであって、１つまたは複数の装置を含む１つまたは複数のノードと、前記１つまたは複数のノードに結合された１つまたは複数の記憶装置と、

前記１つまたは複数のノードによってアクセス可能な高可用性データベースであって、障害が存在する状態で前記１つまたは複数のノードにコヒーレント・データを提供する高可用性データベースとを含み、

前記１つまたは複数の記憶装置に対する前記１つまたは複数の装置のマッピングが前記高可用性データベースに記憶され、

前記マッピングが更新されると、前記高可用性データベースが前記マッピングを更新する前に前記１つまたは複数のノードが前記１つまたは複数の記憶装置へのデータ要求の発行を停止し、前記マッピングが更新されたときに前記１つまたは複数のノードがデータ要求の発行を再開する、分散コンピューティング・システム。

【請求項５０】　前記ノードがデータ要求の発行を停止したときに前記ノードが前記高可用性データベースに肯定応答信号を送る、請求項４９に記載の分散コンピューティング・システム。

【請求項５１】　前記１つまたは複数のノードが、前記肯定応答信号を送る

前に未処理のデータ要求が完了するのを待つ、請求項５０に記載の分散コンピューティング・システム。

【請求項５２】　前記高可用性データベースが、前記肯定応答信号を受け取ったあとで前記マッピングを更新する、請求項５１に記載の分散コンピューティング・システム。

【請求項５３】　前記高可用性データベースが第１の同期信号を出力して、前記マッピングの保留中の更新を示す、請求項５２に記載の分散コンピューティング・システム。

【請求項５４】　前記高可用性データベースが第２の同期信号を出力して、前記マッピングが更新されることを示す、請求項５３に記載の分散コンピューティング・システム。

【請求項５５】　前記第１の同期コマンドと前記第２の同期コマンドが前記１つまたは複数のノードに同時に発行される、請求項５４に記載の分散コンピューティング・システム。

【請求項５６】　前記高可用性データベースが、前記マッピングを更新する前に各アクティブ・ノードからの肯定応答信号を待つ、請求項５５に記載の分散コンピューティング・システム。

【請求項５７】　前記コンピュータ・システムが、データを失うかまたは破損することなしに前記１つまたは複数のノードと前記１つまたは複数の記憶装置との間の前記接続を再構成する、請求項５６に記載の分散コンピューティング・システム。

【請求項５８】　記憶装置に対するノードのマッピングを更新する方法であって、

前記マッピングを高可用性データベースに記憶するステップであって、前記データベースが前記ノードによってアクセス可能であり、障害が存在する状態で前記ノードにコヒーレント・データを提供するステップと、

前記データベースが更新済みマッピングが保留中であるという表示を前記ノードに出力するステップと、

前記ノードが前記記憶装置へのデータ要求を中断するステップと、

前記ノードが未処理のデータ要求が完了するのを待つステップと、

前記ノードが前記マッピングの内部表現を無効にするステップと、

前記ノードが前記データベースに肯定応答信号を出力するステップと、

前記データベースがアクティブ・ノードからの前記肯定応答信号を待つステップと、

前記データベースが前記マッピングを更新するステップと、

前記データベースが前記更新が完了したという表示を前記ノードに出力するステップと、

前記ノードが前記データベースから前記マッピングの更新済みバージョンを要求するステップと、

前記ノードが前記記憶装置への前記データ要求の送信を再開するステップとを含む方法。

【請求項５９】　前記データベースがアクティブ・ノードから肯定応答信号を受け取らない場合、前記データベースが前記ノードに取消し表示を出力して、前記マッピングの前記内部表現を再活動化する、請求項５８に記載の方法。

【請求項６０】　前記ノードへの前記表示が前記データベースからの同期信号である、請求項５８に記載の方法。

【請求項６１】　前記同期信号が前記１つまたは複数のノードに同時に発行される、請求項６０に記載の方法。

【請求項６２】　データを失うかまたは破損することなしに前記マッピングが更新される、請求項６１に記載の方法。

【請求項６３】　記憶装置に対するノードのマッピングを更新する方法であって、

前記マッピングを高可用性データベースに記憶するステップであって、前記データベースが前記ノードによってアクセス可能であり、障害が存在する状態で前記ノードにコヒーレント・データを提供するステップと、

前記データベースが更新済みマッピングが保留中であるという表示を前記ノードに出力するステップと、

前記ノードが前記記憶装置へのデータ要求を中断するステップと

前記データベースが前記マッピングを更新するステップと、

前記データベースが前記更新が完了したという表示を前記ノードに出力するステップと、

前記ノードが前記記憶装置への前記データ要求の送信を再開するステップとを含む方法。

【請求項６４】　前記データベースが前記マッピングを更新する前に、

前記ノードが前記データベースに肯定応答信号を出力するステップと、

前記データベースが前記アクティブ・ノードからの前記肯定応答信号を待つステップとをさらに含む、請求項６３に記載の方法。

【請求項６５】　前記ノードが前記データベースに肯定応答信号を出力する前に、

前記ノードが未処理のデータ要求が完了するのを待つステップと、

前記ノードが前記マッピングの内部表現を無効にするステップとをさらに含む、請求項６４に記載の方法。

【請求項６６】　前記ノードが前記記憶装置への前記データ要求の送信を再開する前に、前記ノードが前記データベースから前記マッピングの更新済みバージョンを要求するステップをさらに含む、請求項６５に記載の方法。

【請求項６７】　前記データベースがアクティブ・ノードから肯定応答信号を受け取らない場合、前記データベースが前記ノードに取消し表示を出力して、前記マッピングの前記内部表現を再活動化する、請求項６３に記載の方法。

【請求項６８】　前記ノードへの前記表示が前記データベースからの同期コマンドである、請求項６７に記載の方法。

【請求項６９】　前記同期信号が前記１つまたは複数のノードに同時に発行される、請求項６８に記載の方法。

【請求項７０】　データを失うかまたは破損することなしに前記マッピングが更新される、請求項６９に記載の方法。

【請求項７１】　記憶装置に対するノードのマッピングを更新するためのプログラム命令を含むコンピュータ可読記憶媒体であって、前記プログラム命令が分散コンピューティング・システムの複数のノード上で実行され、前記プログラ

ム命令が、

前記マッピングを高可用性データベースに記憶するステップであって、前記データベースが前記ノードによってアクセス可能であり、障害が存在する状態で前記ノードにコヒーレント・データを提供するステップと、

前記データベースが更新済みマッピングが保留中であるという表示を前記ノードに出力するステップと、

前記ノードが前記記憶装置へのデータ要求を中断するステップと、

前記ノードが未処理のデータ要求が完了するのを待つステップと、

前記ノードが前記マッピングの内部表現を無効にするステップと、

前記ノードが前記データベースに肯定応答信号を出力するステップと、

前記データベースがアクティブ・ノードからの前記肯定応答信号を待つステップと、

前記データベースが前記マッピングを更新するステップと、

前記データベースが前記更新が完了したという表示を前記ノードに出力するステップと、

前記ノードが前記データベースから前記マッピングの更新済みバージョンを要求するステップと、

前記ノードが前記記憶装置への前記データ要求の送信を再開するステップとを実施するように動作可能である、コンピュータ可読記憶媒体。

【請求項７２】　第１のノードと、第２のノードと、第３のノードと、データ通信リンクとを含む分散コンピューティング・システムのデータ・トランスポート・システムであって、

前記分散コンピューティング・システムのアクティブ・ノードの数と、前記アクティブ・ノード間のリンクの数を決定し、前記リンクにより接続を確立するように構成された構成モジュールと、

前記構成モジュールから前記アクティブ・ノードの数と前記リンクの数を示すデータを受け取り、第１のアクティブ・ノードにデータを転送するための要求をクライアントから受け取り、１つまたは複数の前記リンクを介して前記第１のアクティブ・ノードに前記データを送るように構成された接続モジュールとを含み

前記アクティブ・ノードの数が変更されると、前記構成モジュールが前記変更を前記接続モジュールに通知し、前記接続モジュールは前記クライアントにとってトランスペアレントな前記アクティブノードへの前記接続を再確立するように構成される、データ・トランスポート・システム。

【請求項７３】　前記構成モジュールが高可用性データベースから前記リンクの数を示すデータを受け取る、請求項７２に記載のデータ・トランスポート・システム。

【請求項７４】　前記高可用性データベースが前記ノードのすべてによってアクセス可能であり、各ノードが一貫したデータを受け取る、請求項７３に記載のデータ・トランスポート・システム。

【請求項７５】　ノード障害が存在する状態で前記高可用性データベースが一貫したデータを記憶する、請求項７４に記載のデータ・トランスポート・システム。

【請求項７６】　前記構成モジュールがデーモンである、請求項７２に記載のデータ・トランスポート・システム。

【請求項７７】　前記接続モジュールがカーネル・モジュールである、請求項７６に記載のデータ・トランスポート・システム。

【請求項７８】　前記構成モジュールと前記接続モジュールがプライベート・インタフェースを介して通信する、請求項７２に記載のデータ・トランスポート・システム。

【請求項７９】　前記データ通信リンクが、前記第１のノード上で実行される前記接続モジュールのインスタンスと前記第２のノード上で実行される前記接続モジュールのインスタンスとのインタフェースを提供する、請求項７２に記載のデータ・トランスポート・システム。

【請求項８０】　前記データ通信リンクが前記第１のノードと前記第２のノードとの間の複数の物理リンクを含み、前記構成モジュールが前記複数の物理リンクを１つの仮想リンクとして管理する、請求項７９に記載のデータ・トランスポート・システム。

【請求項８１】　前記データ・トランスポート・システムが複数のクライアントに対応する、請求項７２に記載のデータ・トランスポート・システム。

【請求項８２】　前記複数のクライアントが前記ノード間でメッセージを送受する、請求項８１に記載のデータ・トランスポート・システム。

【請求項８３】　前記構成モジュールがコールバック機能を介して他のアクティブ・ノードから受け取ったメッセージをクライアントに通知する、請求項８２に記載のデータ・トランスポート・システム。

【請求項８４】　前記データ・トランスポート・システムによって転送されたデータがメッセージを含む、請求項７２に記載のデータ・トランスポート・システム。

【請求項８５】　１つのメッセージが制御メッセージとデータ部分の両方を含む、請求項８４に記載のデータ・トランスポート・システム。

【請求項８６】　前記接続モジュールがメッセージ用の記憶空間を割り振り、解放する、請求項７２に記載のデータ・トランスポート・システム。

【請求項８７】　あるメッセージからのデータがもはや不要になったときにクライアントが前記接続モジュールに通知し、前記接続モジュールが前記メッセージに関連する記憶空間を解放する、請求項８６に記載のデータ・トランスポート・システム。

【請求項８８】　複数のノードと１つのデータ通信バスとを含む分散コンピューティング・システム内でデータを移送する方法であって、

前記分散コンピューティング・システム内の物理資源を決定するステップであって、前記物理資源が前記分散コンピューティング・システムのアクティブ・ノードと、前記アクティブ・ノード間のアクティブ・リンクとを含むステップと、

前記アクティブ・リンクにより接続を確立するステップと、

前記アクティブ・ノードのうちの第１のノードにデータを送るためのデータ・アクセス要求を受け取るステップと、

前記アクティブ・リンクのうちの１つまたは複数により前記第１のアクティブ・ノードに前記データを送るステップと、

前記物理資源が変更されたことを決定するステップと、

前記変更済み物理資源への接続を再確立するステップとを含み、

前記変更済み資源の決定と前記リンクの再確立がクライアントにとってトランスペアレントなものである方法。

【請求項８９】　アクティブ・ノード間の複数のリンクが１つの論理リンクとして扱われる、請求項８８に記載の方法。

【請求項９０】　前記物理資源の決定がデーモン・モジュールによって実行される、請求項８８に記載の方法。

【請求項９１】　前記アクティブ・リンクによる接続の確立がデーモン・モジュールによって実行される、請求項９０に記載の方法。

【請求項９２】　前記アクティブ・ノードへの前記データの運搬がカーネル・モジュールによって実行される、請求項９１に記載の方法。

【請求項９３】　複数のクライアントがサポートされ、前記アクティブ・ノードに運搬される前記データが、データ・アクセス要求を要求したクライアントの識別を含む、請求項８８に記載の方法。

【請求項９４】　前記運搬されたデータが制御部分とデータ部分とを含む、請求項８８に記載の方法。

【請求項９５】　アクティブ・ノードに運搬される前記データを記憶するためのメモリ空間を割り振るステップと、

前記メモリ空間を解放するステップとをさらに含む、請求項８８に記載の方法。

【請求項９６】　前記クライアントに向けられるデータの受取りを宛先ノード側のクライアントに通知するステップをさらに含む、請求項８９に記載の方法。

【請求項９７】　物理資源の決定が、物理資源のリストを記憶する高可用性データベースにアクセスすることを含む、請求項８９に記載の方法。

【請求項９８】　前記高可用性データベースが前記アクティブ・ノードによってアクセス可能であり、前記アクティブ・ノードが一貫した構成データを有する、請求項９７に記載の方法。

【請求項９９】　複数のノードと１つのデータ通信リンクとを含む分散コン

ピューティング・システム内でデータを移送するためのプログラム命令を含むコンピュータ可読記憶媒体であって、前記プログラム命令が前記分散コンピューティング・システムの前記複数のノード上で実行され、前記プログラム命令が、

前記分散コンピューティング・システム内の物理資源を決定するステップであって、前記物理資源が前記分散コンピューティング・システムのアクティブ・ノードと、前記アクティブ・ノード間のアクティブ・リンクとを含むステップと、

前記アクティブ・リンクにより接続を確立するステップと、

前記アクティブ・ノードのうちの第１のノードにデータを送るためのデータ・アクセス要求を受け取るステップと、

前記アクティブ・リンクのうちの１つまたは複数により前記第１のアクティブ・ノードに前記データを送るステップと、

前記物理資源が変更されたことを決定するステップと、

前記変更済み物理資源への接続を再確立するステップとを実施するように動作可能であり、

前記変更済み資源の決定と前記接続の再確立がクライアントにとってトランスペアレントなものである、コンピュータ可読記憶媒体。

【請求項１００】　アクティブ・ノード間の複数のリンクが１つの論理リンクとして扱われる、請求項９９に記載のコンピュータ可読記憶媒体。

【請求項１０１】　前記物理資源の決定がデーモン・モジュールによって実行される、請求項９９に記載のコンピュータ可読記憶媒体。

【請求項１０２】　前記アクティブ・リンクによる接続の確立がデーモン・モジュールによって実行される、請求項１０１に記載のコンピュータ可読記憶媒体。

【請求項１０３】　前記アクティブ・ノードへの前記データの運搬がカーネル・モジュールによって実行される、請求項１０２に記載のコンピュータ可読記憶媒体。

【請求項１０４】　アクティブ・ノードに運搬される前記データを記憶するためのメモリ空間を割り振るステップと、

前記メモリ空間を解放するステップとをさらに含む、請求項９９に記載のコン

ピュータ可読記憶媒体。

【請求項１０５】　前記クライアントに向けられるデータの受取りを宛先ノード側のクライアントに通知するステップをさらに含む、請求項９９に記載のコンピュータ可読記憶媒体。

【請求項１０６】　物理資源の決定が、物理資源のリストを記憶する高可用性データベースにアクセスすることを含む、請求項１００に記載のコンピュータ可読記憶媒体。

【請求項１０７】　前記高可用性データベースが前記アクティブ・ノードによってアクセス可能であり、前記アクティブ・ノードが一貫した構成データを有する、請求項１０６に記載のコンピュータ可読記憶媒体。

【発明の詳細な説明】

【０００１】

（発明の背景）

（１．発明の分野）

本発明は、分散コンピューティング・システムの分野に関し、より詳細には分散仮想記憶装置に関する。

【０００２】

（２．関連技術の説明）

クラスタなどの分散コンピューティング・システムは２つまたはそれ以上のノードを含むことがあり、それらのノードはコンピューティング・タスクを実行するために使用される。一般的に言えば、ノードとは、１つまたは複数のコンピューティング・タスクを実行するように設計された１つの回路グループである。１つのノードは、１つまたは複数のプロセッサと、１つのメモリと、インタフェース回路とを含むことができる。一般的に言えば、クラスタとは、ノード間でデータを交換する能力を有する２つまたはそれ以上のノードからなるグループである。あるノードで特定のコンピューティング・タスクを実行することができるが、他のノードは無関係のコンピューティング・タスクを実行する。あるいは、特定のコンピューティング・タスクを全体として実行するのに必要な時間を削減するために、そのコンピューティング・タスクのコンポーネントをノード間に分散することができる。一般的に言えば、プロセッサとは、１つまたは複数のオペランドの演算を実行して結果を生成するように構成されたデバイスである。演算は、プロセッサによって実行される命令に応答して実行することができる。

【０００３】

１つのクラスタ内のノードは、そのノードに結合された１つまたは複数の記憶装置を有することができる。一般的に言えば、記憶装置とは、大量のデータを格納することができる持続性装置である。たとえば、記憶装置は、ディスク装置などの磁気記憶装置またはコンパクト・ディスク装置などの光学記憶装置にすることができる。ディスク装置は記憶装置の一例に過ぎないが、「ディスク」という用語は本明細書全体を通して「記憶装置」と交換して使用することができる。記

憶装置に物理的に接続されたノードは記憶装置に直接アクセスすることができる。記憶装置はあるクラスタの１つまたは複数のノードに物理的に接続することができるが、その記憶装置はあるクラスタのすべてのノードに物理的に接続できるわけではない。ある記憶装置に物理的に接続されていないノードはその記憶装置に直接アクセスすることはできない。クラスタによっては、記憶装置に物理的に接続されていないノードは、ノード同士を接続するデータ通信リンクを介してその記憶装置に間接的にアクセスすることができる。

【０００４】

あるノードがクラスタ内のどの記憶装置にもアクセスできる、すなわちそのノードに記憶装置が物理的に接続されているかのようにすることは有利である。たとえば、Ｏｒａｃｌｅ　Ｐａｒａｌｌｅｌ　Ｓｅｒｖｅｒなどの一部のアプリケーションでは、ＵＮＩＸデバイス・セマンティクスにより１つのクラスタ内のすべての記憶装置にアクセスしなければならない場合がある。あるノードに物理的に接続されていないが、あるノードに物理的に接続されているように見える記憶装置は、仮想装置または仮想ディスクと呼ばれる。一般的に言えば、分散仮想ディスク・システムとは、２つまたはそれ以上のノード上で動作するソフトウェア・プログラムであり、クライアントと１つまたは複数の記憶装置とのインタフェースとなり、その１つまたは複数の記憶装置がそのノードに直接接続されているというように見えるソフトウェア・プログラムである。一般的に言えば、クライアントとは、あるアクションを開始するためにプログラムにアクセスするプログラムまたはサブルーチンである。クライアントは、アプリケーション・プルグラムである場合もあれば、オペレーティング・システム・サブルーチンである場合もある。

【０００５】

残念ながら、従来の仮想ディスク・システムは、一貫した仮想ディスク・マッピングを保証していない。一般的に言えば、記憶装置マッピングは、ある記憶装置がどのノードに物理的に接続されるか、ならびにこれらのノード上のどのディスク装置がその記憶装置に対応するかを識別するものである。ある仮想装置をある記憶装置にマッピングするノードとディスク・デバイスは、ノード／ディスク

対と呼ぶ場合もある。仮想装置マッピングは、許可およびその他の情報を含むこともある。ノード障害などの障害が発生した場合にマッピングが持続性のものであることが望ましい。あるノードが他のノードの支援なしにある装置と通信できる場合、そのノードはその装置に物理的に接続される。

【０００６】

クラスタは、ボリューム・マネージャを実装することができる。ボリューム・マネージャとは、そのクラスタの記憶資源を管理するツールである。たとえば、ボリューム・マネージャは、２つの記憶装置をミラーして１つの高可用性ボリュームを作成することができる。他の実施形態では、ボリューム・マネージャは、複数の記憶装置にわたるファイルの部分を記憶するストライピングを実施することができる。従来の仮想ディスク・システムは、記憶装置の上下いずれかに層状に重ねたボリューム・マネージャをサポートすることができない。

【０００７】

他の望ましい特徴としては、ノード障害または記憶装置経路障害などの障害が存在する状態でデータ・アクセス要求が確実に実行されるような高可用性のデータ・アクセス要求を含む。一般的に言えば、記憶装置経路とは、あるノードからある記憶装置への直接接続部である。一般的に言えば、データ・アクセス要求とは、データを読み書きするための記憶装置への要求である。

【０００８】

仮想ディスク・システムでは、複数のノードが１つの記憶装置の多くの表現を持つことがある。残念ながら、従来のシステムは、各ノード上の表現が一貫した許可データを有することを保証する確実な手段を提供しない。一般的に言えば、許可データは、どのユーザが装置、ディレクトリ、またはファイルにアクセスするための許可を有するかを識別するものである。許可としては、読取り許可、書込み許可、実行許可を含むことができる。

【０００９】

さらに、あるクラスタのノードを追加または除去する能力を有するか、またはそのクラスタが動作している間に既存のノードと記憶装置との接続を変更することが望ましい。この能力は、そのクラスタを低下させることができないクリティ

カル・アプリケーションで使用するクラスタでは特に重要である。この能力により、物理的資源（ノードと記憶装置など）をシステムに追加するか、あるいはクラスタ内のデータ・アクセス要求を損なわずに修理および交換を実施することができる。

【００１０】

（発明の概要）

上記で概要を示した問題は、本発明による高可用性仮想ディスク・システムによって大部分が解決される。一実施態様における高可用性仮想ディスク・システムは、各記憶装置とクラスタ内の各ノードとのインタフェースを提供する。ノードの見地からすると、各記憶装置がそのノードに物理的に接続されているとみることができる。あるノードがある記憶装置に物理的に接続されている場合、仮想ディスク・システムはその記憶装置に直接アクセスできる。あるいは、そのノードがある記憶装置に物理的に接続されていない場合、仮想ディスク・システムはそのクラスタ内にあってその記憶装置に物理的に接続されている他のノードを通ってその記憶装置にアクセスできる。一実施態様では、すべてのノードが１つのデータ通信リンクを介して通信する。ある記憶装置が直接アクセスされるかまたは他のノードを介してアクセスされるかは、その記憶装置にアクセスするクライアントにとってトランスペアレントなものである。

【００１１】

一実施態様では、ノードは仮想ディスクのマッピングを記憶装置に記憶する。たとえば、各アクティブ・ノードは、各仮想装置用の一次ノード／ディスク対と二次ノード／ディスク対を識別するマッピングを記憶することができる。各ノード／ディスク対は、その記憶装置に物理的に結合されたノードと、その記憶装置に対応するそのノード上のディスク装置を識別する。二次ノード／ディスク対は、代替ノード／ディスク対ともいう場合がある。そのノードが一次ノード／ディスク対を介してある記憶装置にアクセスできない場合、そのノードは二次ノード／ディスク対を介してデータ・アクセス要求を再試行することができる。障害が存在する状態でノード間の一貫したマッピングを維持するために、そのマッピングを高可用性データベースに記憶することができる。高可用性データベースは障

害が存在する状態でも１つの一貫したデータ・コピーを維持するので、高可用性データベースに照会する各ノードは、同じマッピングを取得することになる。また、高可用性データベースを使用して、仮想装置へのアクセスを制御するための許可データを記憶することもできる。高可用性データベースは障害が存在する状態でも１つの一貫した許可データ・コピーを維持するので、そのデータベースに照会する各ノードは、同じ許可データを取得することになる。

【００１２】

本発明による仮想ディスク・システムの特徴の１つは、システムの高い可用性である。一実施態様では、仮想ディスク・システムは、それが受け取ったすべてのデータ・アクセス要求を記憶し、エラーが発生した場合にその要求を再試行する。たとえば、データ・アクセス要求を開始し、要求側ノードと呼ばれるノードの仮想ディスク・システムは、すべての未処理のデータ要求を記憶することができる。宛先ノード、すなわち、そのデータ・アクセス要求が向けられるノードがそのデータ・アクセス要求を完了できない場合、要求側ノードに対してエラー表示を返すことができ、要求側ノードはその記憶装置に接続されている代替ノードにそのデータ・アクセス要求を再送することができる。このエラー検出および再試行は自動的に実行され、クライアントにとってトランスペアレントなものである。他の例では、ノード障害が発生した場合、仮想ディスク・システムは、アクティブ・ノードの変更済みリストを受け取り、その記憶装置に結合されたアクティブ・ノードに不完全なデータ・アクセス要求を再送することができる。この再構成および再試行もクライアントにとってトランスペアレントなものである。

【００１３】

本発明による仮想ディスク・システムの他の特徴は、クラスタが動作している間にそのクラスタを再構成できる能力である。あるクラスタを再構成すると、記憶装置に対する仮想ディスクのマッピングを更新することができる。エラーを防止するため、マッピングを更新する前にそのクラスタのすべてのノードに対して同期コマンドを実行または操作することができる。この同期コマンドにより、ノードはデータ・アクセス要求の発行を停止する。マッピングを更新した後、他の同期コマンドにより、そのノードはデータ・アクセス要求の発行を再開する。

【００１４】

仮想ディスク・システムは、ボリューム・マネージャと記憶装置とのインタフェースまたはクライアントとボリューム・マネージャとのインタフェースとして機能するように設計することができる。前者の構成では、クライアントはボリューム・マネージャにインタフェースし、ボリューム・マネージャは仮想ディスク・システムにインタフェースする。後者の構成では、クライアントは仮想ディスク・システムにインタフェースし、仮想ディスク・システムはボリューム・マネージャにインタフェースする。

【００１５】

本発明の他の目的および利点は、以下に示す詳細説明を読み、添付図面を参照すると明らかになるだろう。

【００１６】

本発明は様々な変更および代替形式が可能であるが、その具体的な実施形態を例証として添付図面に示し、本明細書に詳細に説明する。しかし、図面およびそれに対する詳細な説明は開示した特定の形式に本発明を限定するためのものではなく、むしろ、その意図は特許請求の範囲によって定義した本発明の精神および範囲に含まれるすべての変更態様、同等態様、代替態様を含むことである。

【００１７】

（発明の詳細な説明）

次に図１に移行すると、本発明の一実施形態によるクラスタ構成のブロック図が示されている。クラスタ１００は、データ通信リンク１０２と、３つのノード１０４Ａ～１０４Ｃと、３つの記憶装置１０８、１１０、１１２とを含む。データ通信リンク１０２は、ノード間でデータを転送するためのデータ通信経路である。データ通信リンク１０２は、マルチドロップ・リンクまたはポイントツーポイント・リンクを企図している。たとえば、データ通信リンク１０２は３つのポイントツーポイント・リンクを含むことができる。第１のリンクはノード１０４Ａと１０４Ｂとの間の通信経路で、第２のリンクはノード１０４Ａと１０４Ｃとの間の通信経路で、第３のリンクはノード１０４Ｂと１０４Ｃとの間の通信経路である。一実施形態のデータ通信リンク１０２はスケーラブル・コヒーレント・

インタフェース（scalable coherent interface:ＳＣＩ）を実装する。特定の一実施形態のクラスタは、ＳＣＩによりデータを転送するためにＴＣＰ／ＩＰプロトコルを実装している。例示のためにのみ３つのノードを示していることに留意されたい。他の実施形態ではそれより多いかまたは少ないノードを使用することもできる。

【００１８】

図示の実施形態では、記憶装置１０８がノード１０４Ａに物理的に接続され、記憶装置１１０がノード１０４Ｂに物理的に接続され、記憶装置１１２がノード１０４Ｃに物理的に接続されている。記憶装置１０８～１１２は一般に、それが接続されているノードのメモリの記憶容量を上回る記憶容量を有する。データは、ノードによって現在使用されていない記憶装置１０８～１１２に記憶され、その記憶装置からのデータがそのデータが必要になったときにそのノードのメモリに記憶またはキャッシュされる。図示の実施形態では、記憶装置が１つのノードのみに物理的に接続されている。代替実施形態では、１つの記憶装置を複数のノードに物理的に接続することができる。複数の物理接続により、ある記憶装置に物理的に接続された１つのノードが故障した場合または記憶装置経路が故障した場合でもその記憶装置にアクセスすることができる。

【００１９】

同じ分散プログラムの複数のインスタンスが各ノード上で動作することができる。たとえば、ボリューム・マネージャ１０５Ａとボリューム・マネージャ１０５Ｂは同じ分散ボリューム・マネージャ・プログラムの異なるインスタンスである。これらのインスタンスは、データ通信リンク１０２を介して互いに通信することができる。各インスタンスには、同じ参照番号とそれに続く固有に英字、たとえば、１０５Ａまたは１０５Ｂが付与される。簡潔にするため、分散プログラムは、まとめて参照番号のみを使用する、たとえば、ボリューム・マネージャ１０５ということができる。

【００２０】

ノード１０４Ａは、ボリューム・マネージャ１０５Ａと仮想ディスク・システム１０６Ａとを含む。図示の実施形態の仮想ディスク・システム１０６Ａは、ボ

リューム・マネージャ１０５と記憶装置１０８～１１２とのインタフェースとなっている。ボリューム・マネージャ１０５Ａの見地からすると、各記憶装置はノード１０４Ａに物理的に接続されているように見える。仮想ディスク・システム１０６は複数のノード上で動作する分散プログラムである。図示の実施形態では、仮想ディスク・システム１０６の１つのインスタンスが各ノードで動作している。仮想ディスク・システム１０６Ａは、ノード１０４Ａで動作する仮想ディスク・システム１０６のインスタンスであり、記憶装置１０８～１１２をそれぞれ表す３つの仮想装置（ＶＤ１、ＶＤ２、ＶＤ３）を含む。ボリューム・マネージャ１０５は、自身のノードに物理的に接続された記憶装置に伝達するのと同じように仮想装置に伝達する。一実施形態では、ボリューム・マネージャ１０５はＵＮＩＸデバイス・ドライバ・セマンティクスを使用する。記憶装置１０８（すなわち、ＶＤ１）へのデータ・アクセス要求は仮想ディスク・システム１０６Ａから記憶装置１０８に直接運搬される。記憶装置１１０および１１２（すなわち、ＶＤ２およびＶＤ３）へのデータ・アクセス要求はデータ通信リンク１０２によりこれらの装置に物理的に接続されたそれぞれのノードに運搬される。

【００２１】

各ノードの仮想ディスクは別個の装置であることに留意されたい。たとえば、ノード１０４Ａ、１０４Ｂ、１０４ＣのＶＤ１は、それぞれ固有のデバイス・ドライバによって管理される固有の装置である。装置は固有であるが、各ＶＤ１装置は同じ物理記憶装置にマッピングする。換言すれば、ノード１０４ＡのＶＤ１にデータを書き込むことは、ノード１０４Ｂまたは１０４ＣのＶＤ１にデータを書き込むのと同様に、記憶装置１０８にデータを記憶する。各記憶装置が複数のノードに物理的に接続できることにさらに留意されたい。この場合、その装置に物理的に接続された各ノードは、記憶装置にインタフェースする異なるデバイス・ドライバを有する。

【００２２】

図示の実施形態では、ボリューム・マネージャ１０５Ａのボリューム１（Ｖ１）がＶＤ１およびＶＤ２に結合されている。一実施形態では、ボリューム・マネージャ１０５Ａがこれらの装置をミラーすることもできる。代替実施形態では、

ボリューム・マネージャ１０５Ａが他の仮想装置に結合された他のボリュームを含むこともできる。たとえば、第２のボリューム・マネージャ１０５ＡはＶＤ２およびＶＤ３に結合することができる。

【００２３】

ノード１０４Ｂおよび１０４Ｃでは、ボリューム・マネージャ（１０５Ｂおよび１０５Ｃ）および仮想ディスク・システム（１０６Ｂおよび１０６Ｃ）がボリューム・マネージャ１０５Ａおよび仮想ディスク・システム１０６Ａと実質的に同じように動作する。図示の実施形態では、ボリューム・マネージャ１０５Ｂのボリューム２（Ｖ２）が仮想ディスク・システム１０６ＢのＶＤ２およびＶＤ３に結合されている。仮想ディスク・システム１０６Ｂは、記憶装置１１０に直接アクセスし、通信インタフェース１０２およびノード１０４Ｃを介して記憶装置１１２にアクセスする。ボリューム・マネージャ１０５Ｃのボリューム３（Ｖ３）は仮想ディスク・システム１０６ＣのＶＤ２およびＶＤ３に結合されている。仮想ディスク・システム１０６Ｃは、記憶装置１１２に直接アクセスし、通信インタフェース１０２およびノード１０４Ｂを介して記憶装置１１０にアクセスする。

【００２４】

次に図２に移行すると、本発明の一実施形態による代替クラスタ構成のブロック図が示されている。クラスタ２００は、データ通信リンク１０２と、３つのノード１０４Ａ～１０４Ｃと、３つの記憶装置１０８、１１０、１１２とを含む。簡潔にするため、図１の構成要素と同様の構成要素には同じ参照番号が付与されている。図２においては、クライアントは、ボリューム・マネージャ１０５ではなく仮想ディスク・システム１０６にインタフェースする。仮想ディスク・システムがボリューム・マネージャにインタフェースし、ボリューム・マネージャは１つまたは複数の記憶装置にインタフェースする。この構成では、ボリューム・マネージャ１０５は仮想ディスク・システム１０６の下に層状に重ねられている。簡潔にするため、ノード１０４Ａの動作についてのみ以下に説明する。ノード１０４Ｂおよび１０４Ｃは実質的に同じように動作する。

【００２５】

ノード１０４Ａでは、クライアントは仮想ディスク・システム１０６Ａにインタフェースする。クライアントの見地からすると、仮想ディスク・システム１０６Ａは３つの別々の記憶装置として現れる。図２の３つの仮想装置は、ボリューム・マネージャが仮想ディスク・システムの下に層状に重ねられていることを反映するように仮想ボリューム（ＶＶ１、ＶＶ２、ＶＶ３）として表示されている。クライアントの見地からすると、仮想ボリュームは記憶装置のように動作する。たとえば、仮想ボリュームはＵＮＩＸデバイス・ドライバ・セマンティクスを使用することができる。クライアントは、仮想ディスク・システム１０６Ａからクラスタの３つのボリュームのいずれにもアクセスすることができる。ボリューム・マネージャ１０５Ａは記憶装置にインタフェースする。図示の実施形態では、ボリューム・マネージャ１０５Ａのボリューム１（Ｖ１）が記憶装置１０８および１１０に結合されている。一実施形態では、ボリューム１は記憶装置１０８および１１０にデータをミラーすることができる。仮想ディスク・システム１０６Ａの見地からすると、ボリューム・マネージャ１０５Ａのボリューム１は記憶装置のように動作する。たとえば、そのボリュームはＵＮＩＸデバイス・ドライバのように動作することができる。

【００２６】

仮想ディスク・システム１０６Ｂの仮想ボリューム２（ＶＶ２）はボリューム・マネージャ１０５Ｂのボリューム２（Ｖ２）に直接インタフェースする。仮想ボリューム１および３は、データ通信リンク１０２を介してノード１０４Ａのボリューム１およびノード１０５Ｃのボリューム３と通信する。同様に、仮想ディスク・システム１０６Ｃの仮想ボリューム３はボリューム・マネージャ１０５Ｃのボリューム３に直接インタフェースする。仮想ボリューム１および２は、データ通信リンク１０２を介してノード１０４Ａのボリューム１およびノード１０５Ｂのボリューム２と通信する。図示の実施形態では、ボリューム・マネージャ１０５Ｂのボリューム２およびボリューム・マネージャ１０５Ｃのボリューム３はどちらも記憶装置１１０および１１２に物理的に接続されている。

【００２７】

ボリューム・マネージャと仮想ディスク・システムはどちらも記憶装置のよう

に動作するので、ボリューム・マネージャは仮想ディスク・システムの上または下のいずれかに層状に重ねることができる。したがって、それがボリューム・マネージャにインタフェースするかまたは仮想ディスク・システムにインタフェースするかはクライアントにとってトランスペアレントなものである。どちらの実施形態でも、クライアントは３つの信頼できる記憶装置に直接アクセスすることができるように見える。ボリューム・マネージャと仮想ディスク・システムはどちらも記憶装置に直接インタフェースすることができる。ボリューム・マネージャによっては、仮想ディスク装置の上に層状に重ねられたときにより良好に動作できるものもある。たとえば、ベリタスＣＶＭなどのクラスタ・ボリューム・マネージャは仮想ディスク・システムの上に層状に重ねられたときに最も良好に動作するが、ソルスティス・ディスク・スイート（ＳＤＳ）などの非分散ボリューム・マネージャは仮想ディスク・システムの下で動作しなければならない場合もある。ボリューム・マネージャは仮想ディスク・システムの下で動作するために分散しなければならないことに留意されたい。仮想ディスク・システムがそれらが１つの装置であるかのようにノードの仮想ディスクを管理するのと同様に、ＣＶＭなどの分散ボリューム・マネージャは、それらが１つのボリュームであるかのようにボリューム（Ｖ１、Ｖ２、Ｖ３）を管理できることにさらに留意されたい。

【００２８】

次に図３に移行すると、本発明の一実施形態によるクラスタの２つのノードで動作する仮想ディスク・システムのブロック図が示されている。図示の実施形態では、各ノードはユーザ部分とカーネルとを含む。ノード１０４Ａのユーザ部分は、クラスタ・メンバシップ・モニタ（ＣＭＭ）３１０Ａと、クラスタ構成データベース（ＣＣＤ）３１１Ａと、クライアント３１２Ａと、ネットディスク・デーモン（ＮＤＤ）３１４Ａと、クラスタ・トランスポート・インタフェース・デーモン（ＣＴＩＤ）３１６Ａとを含む。ノード１０４Ａのカーネルは、ネットディスク・ドライバ（ＮＤ）３１８Ａと、ネットディスク・マスタ（ＮＭ）３２０Ａと、クラスタ・トランスポート・インタフェース（ＣＴＩ）３２２Ａ、クラスタ接続性モニタ（ＣＣＭ）３２４Ａと、ディスク・ドライバ３２６Ａと、ネット

ワーク・トランスポート３２８Ａとを含む。ノード１０４Ｂのユーザ部分は、クラスタ・メンバシップ・モニタ（ＣＭＭ）３１０Ｂと、クラスタ構成データベース（ＣＣＤ）３１１Ｂと、ネットディスク・デーモン（ＮＤＤ）３１４Ｂと、クラスタ・トランスポート・インタフェース・デーモン（ＣＴＩＤ）３１６Ｂとを含む。ノード１０４Ｂのカーネルは、ネットディスク・ドライバ（ＮＤ）３１８Ｂと、ネットディスク・マスタ（ＮＭ）３２０Ｂと、クラスタ・トランスポート・インタフェース（ＣＴＩ）３２２Ｂ、クラスタ接続性モニタ（ＣＣＭ）３２４Ｂと、ネットディスク・ドライバ３２６Ｂと、ネットワーク・トランスポート３２８Ｂとを含む。

【００２９】

図示の実施形態ではボリューム・マネージャが含まれていない。図１および図２に関連して前述したように、ボリューム・マネージャは仮想ディスク・システムの上または下のいずれかに実装することができる。ボリューム・マネージャが仮想ディスク・システムの上に実装される場合、クライアント３１２Ａがボリューム・マネージャにインタフェースし、次にそのボリューム・マネージャがＮＤ３１８Ａにインタフェースする。反面、ボリューム・マネージャが仮想ディスク・システムの下に実装される場合、ＮＭ３２０Ａがボリューム・マネージャにインタフェースし、次にそのボリューム・マネージャがディスク・ドライバ３２６Ａにインタフェースする。

【００３０】

ＣＴＩＤ３１６Ａという構成モジュールは、ＣＴＩ３２２Ａという接続モジュールを初期設定するデーモンである。クラスタの構成が変更されると、ノード３１６Ａは初期設定される。ＣＴＩＤ３１６ＡはＣＣＤ３１１Ａに照会して構成情報を入手する。一実施形態の構成情報は、そのクラスタのノード間のリンクの数と、リンクに関連するプロトコルとを示す。一実施形態では、ＣＴＩＤ３１６ＡがＣＭＭ３１０Ａをさらに照会し、クラスタ内のアクティブ・ノードのリストなどのメンバシップ情報を入手する。ＣＴＩＤ３１６Ａは、ノード間のリンクにより接続を確立し、メンバシップ情報とリンク情報をＣＴＩ３２２Ａに送る。ＣＴＩＤ３１６Ａは、プライベート相互接続によりＣＴＩ３２２Ａに連絡することが

でき、入出力制御要求を使用することができる。

【００３１】

ＣＣＤ３１１Ａによって識別されるリンクは、物理リンクの場合もあれば、仮想リンクの場合もある。たとえば、ＣＣＭ３２４Ａは、ＣＴＩ３２２Ａによってアクセス可能な１つの仮想リンクとして一対の物理リンクを管理することができる。ＣＣＭ３２４については図９に関連して以下に詳述する。

【００３２】

ＣＣＤ３１１Ａは、分散高可用性クラスタ・データベースのインスタンスの１つである。ＣＣＤ３１１は障害が存在する状態でも一貫したデータを記憶する。ＣＣＤ３１１にマッピング・データを記憶することにより、各ノードは障害が存在する状態でも同じマッピング情報を入手する。ＣＣＤ３１１については、Ｓｌａｕｇｈｔｅｒ他により１９９７年１０月２１日に出願され、「Ｈｉｇｈｌｙ ａｖａｉｌａｂｌｅ Ｄｉｓｔｒｉｂｕｔｅｄ Ｃｌｕｓｔｅｒ Ｃｏｎｆｉｇｕｒａｔｉｏｎ Ｄａｔａｂａｓｅ」という名称で本願譲受人に譲渡された同時係属特許出願第０８／９５４７９６号に詳述されている。

【００３３】

ＣＭＭ３１０は、クラスタ・メンバシップを監視する分散プログラムである。メンバシップが変更されると、ＣＭＭ３１０はその変更を検出し、ＣＴＩＤ３１６ＡおよびＮＤＤ３１４Ａなどクラスタ内の他の資源に新しいメンバシップ情報を送る。メンバシップ変更の例としては、そのクラスタに加わるノードまたはそのクラスタを離れるノードを含む。一実施形態のＣＭＭ３１０は各構成に固有の構成番号を出力する。

【００３４】

ＮＤＤ３１４Ａは、新しい装置をオープンしたときまたは再構成中にＮＤ３１８Ａを初期設定するデーモンである。再構成は、ノードがそのクラスタに加わったときまたはそのクラスタを離れるとき、あるいはノードが故障したときに行われる。一実施形態では、各仮想ディスク装置は個別に初期設定される。特定の一実施形態の仮想ディスク装置は、そのクラスタがその装置をオープンしたときにクラスタによって初期設定されるか、または再構成の前に仮想ディスク・ドライ

バをオープンした場合は再構成後にクラスタによって初期設定される。このため、すべての仮想ディスク装置がそれぞれの再構成後に初期設定されるわけではない。

【００３５】

一実施形態のＮＤ３１８Ａは、オープンすべき装置のリストと、オープンした装置のリストを記憶する。クライアントがある装置をオープンするよう要求すると、ＮＤ３１８Ａはオープンすべき装置のリストにその装置を追加する。ＮＤＤ３１４Ａはオープンすべき装置のリストに照会する。そのリストがオープンすべき装置を含む場合、ＮＤＤ３１４ＡはＣＣＤ３１１Ａに照会し、識別した装置に関するマッピング情報を入手する。ＮＤＤ３１４Ａは、ＣＭＭ３１０Ａにも照会して、アクティブ・ノードのリストなどのメンバシップ情報を入手することもできる。ＮＤＤ３１４Ａは、マッピング情報とメンバシップ情報をＮＤ３１８Ａに送る。ＮＤＤ３１４Ａはプライベート相互接続によりＮＤ３１８Ａに連絡することができ、入出力制御要求を使用することができる。

【００３６】

一実施形態では、ある装置に関するマッピング情報は、ある記憶装置に物理的に接続された一次および二次ノードと、その記憶装置に対応するこれらのノードのディスク装置とを識別する。ノードとディスクの各対はノード／ディスク対ともいう場合がある。一次および二次ノード／ディスク対とメンバシップ情報とに基づいて、ＮＤ３１８Ａは、ある装置に関するデータ・アクセス要求を経路指定するためのノードを選択することができる。ＮＤ３１４ＡとＣＴＩ３２２Ａが初期設定されると、仮想ディスク・システムはクライアント３１２Ａからデータ・アクセス要求を受け入れる準備が整っている。

【００３７】

クライアント３１２Ａは、それが記憶装置にアクセスするのと同じように仮想ディスク・システムの仮想装置にアクセスする。クライアントの見地からすると、各記憶装置またはボリュームはそのノードに物理的に接続されているように見える。図示の実施形態では、クライアント３１２Ａがある記憶装置からのデータにアクセスする場合、クライアントはデータ・アクセス要求をＮＤ３１８Ａに送

る。一実施形態では、クライアント３１２Ａは、宛先記憶装置と、動作のタイプと、データを検索または記憶するための位置とをＮＤ３１２Ａに対して指定する。残りの動作はクライアント３１２Ａにとってトランスペアレントなものになる。ＮＤ３１８Ａは、マッピングおよび現行メンバシップ情報に基づいて、どのノードにデータ・アクセス要求を送るかを決定する。一実施形態では、ＣＣＤ３１１Ａから入手したマッピング情報は、その記憶装置に物理的に接続された一次および二次ノードを含む。ＮＤ３１８Ａは、一次ノードがアクティブである場合、そのデータ・アクセス要求を一次ノードに経路指定することができる。あるいは、一次ノードがアクティブではない場合、ＮＤ３１８Ａはそのデータ・アクセス要求を二次ノードに経路指定する。その記憶装置にアクセスするためにどのノードを使用するかは、クライアント３１２Ａにとってトランスペアレントなものになる。

【００３８】

ＮＤ３１８Ａは、ＣＴＩ３２２Ａにデータ・アクセス要求を送り、どのノードにデータ・アクセス要求を送るかを指定する。ＣＴＩ３２２Ａがどのようにデータ・アクセス要求を宛先ノードに転送するかは、ＮＤ３１８Ａおよびクライアント３１２Ａにとってトランスペアレントなものになる。一実施形態では、その記憶装置がノード１０４Ａに直接結合されている場合、ＮＤ３１８ＡはＣＴＩ３２２Ａではなく ＮＭ３２０Ａにデータ・アクセス要求を送る。ＮＭ３２０Ａはデータ・アクセス要求をディスク・ドライバ３２６Ａに送り、次にそのディスク・ドライバはその記憶装置にアクセスする。一実施形態のＮＭ３２０Ａは、ＮＤ３１８Ａのうち、ディスク・ドライバ３２６Ａにインタフェースする部分である。ディスク・ドライバ３２６Ａは、ノード１０４Ａに物理的に接続された１つまたは複数の記憶装置にインタフェースする。

【００３９】

ＣＴＩ３２２Ａは複数のリンクを管理する。ＣＴＩ３２２Ａは分散プログラムＣＴＩ３２２のインスタンスの１つである。ＣＴＩ３２２Ａは、あるデータ・アクセス要求の宛先ノードへの１つまたは複数のリンクを管理することができる。たとえば、そのデータ・アクセス要求の宛先ノードがノード１０４Ｂである場合

、ＣＴＩ３２２Ａはそのノードへの３つのリンクを管理することができる。ＣＴＩ３２２Ａは、１つのリンクを介してノード１０４Ｂにすべてのデータを移送する場合もあれば、３つのリンクによりデータを分散する場合もある。ＣＴＩ３２２Ａは、宛先ノードで宛先クライアントを識別するためのフィールドをデータ・アクセス要求に付加することができる。ノード１０４ＢのＣＴＩ３２２Ｂは複数のクライアントに対応することができる。ＣＴＩ３２２Ａによってメッセージに付加されたフィールドは、ＣＴＩ３２２Ｂがどのクライアントにそのデータを経路指定するべきかを識別するものである。たとえば、ＣＴＩ３２２Ａは、宛先クライアントをＮＤ３１８Ｂとして指定するデータを、ＮＤ３１８Ａが受け取るデータ要求に付加することができる。

【００４０】

一実施形態のＣＣＭ３２４Ａは、２つまたはそれ以上の冗長物理リンクを管理する。ＣＴＩ３２２Ａの見地からすると、冗長物理リンクは１つの論理リンクとして現れる。ＣＣＭ３２４Ａは、物理リンクによりＣＣＭ３２４Ｂとメッセージを交換する。ＣＣＭ３２４の２つのインスタンスは、冗長リンクのうちのどちらが動作可能であるかに関して合意に達する。ＣＭＭ３２４は、データを転送するために１つの動作可能な物理リンクを選ぶことができる。そのリンクが故障した場合、ＣＣＭ３２４は、その故障を検出し、代替リンク上でデータを転送することができる。ＣＴＩ３２２の見地からすると、各論理リンクは１つの高可用性リンクとして現れる。一実施形態のＣＣＭ３２４Ａはそのクラスタ内の各ノードへのリンクを管理する。たとえば、ＣＭＭ３２４Ａはノード１０４Ｂおよび１０４Ｃへのリンクを管理することができる。

【００４１】

ネットワーク・トランスポート３２８Ａは、データ通信リンク１１２のリンクによりプロトコル機能を実行する。一実施形態では、データ通信リンク１１２によりＴＣＰ／ＩＰプロトコルを使用する。他の実施形態では、他のプロトコルを実装することができる。たとえば、低待ち時間接続性層（ＬＬＣＬ）、メッセージ受渡しインタフェース（ＭＰＩ）、低オーバヘッド通信（ＬＯＣＯ）などの高速プロトコルを使用することができる。

【００４２】

ノード１０４Ｂでは、ネットワーク・トランスポート３２８Ｂがデータ・アクセス要求を受け取り、適切なプロトコルを使用してデータをＣＴＩ３２２Ｂに移送する。ＣＴＩ３２２Ｂは、データ・アクセス要求を部分的にデコードし、その宛先クライアントを決定することができる。図示の実施形態では、データはＮＤ３１８Ｂに経路指定される。ＮＤ３１８Ｂは、データ・アクセス要求を部分的にデコードし、宛先記憶装置を決定することができる。その記憶装置がノード１０４Ｂに物理的に結合されている場合、ＮＤ３１８Ｂは要求をＮＭ３２０Ｂに送り、ＮＭ３２０Ｂはその要求をディスク・ドライバ３２６Ｂに送る。ディスク・ドライバ３２６Ｂはその記憶装置にアクセスする。データ・アクセス要求が読取りトランザクションである場合、要求されたデータはＮＤ３１８、ＣＴＩ３２２、データ通信リンク１１２を介してクライアント３１２Ａに戻される。

【００４３】

本発明の一実施形態による仮想ディスク・システムの特徴の１つは高い可用性である。この仮想ディスク・システムは、ノード障害などの障害が存在する状態でデータ・アクセス要求が確実に実行されるように設計されている。この目的に向かって、ＮＤ３１８Ａは保留データ・アクセス要求のリストを記憶する。データ・アクセス要求が正常に完了しない場合、仮想ディスク・システムは他のノードへのデータ・アクセス要求を再試行する。要求側ノードは、否定応答信号を受け取ることによって不完全なデータ・アクセス要求を検出する場合もあれば、宛先ノードがアクティブではないことを示す再構成データを受け取る場合もある。データ・アクセス要求が正常に完了した場合、それは保留データ・アクセス要求のリストから除去される。

【００４４】

たとえば、ノード１０４Ｂがある記憶装置の一次ノードであり、ノード１０４Ｃがその記憶装置の二次ノードである場合が考えられる。ＮＤ３１８Ａがその記憶装置にデータ・アクセス要求を送ると、それはそのデータ・アクセス要求を一次ノードに送ることができ、そのノードはノード１０４Ｂである。ノード１０４Ｂがデータ・アクセス要求を正常に完了できない場合、たとえば、ディスク・ド

ライバ３２６Ｂと記憶装置との間の記憶装置経路が機能しない場合、ノード１０４Ａは、そのデータ・アクセス要求が正常に完了しなかったことを示す否定応答信号を受け取ることができる。次にノード１０４Ａはデータ・アクセス要求を二次ノードに再送することができる。そのノードはノード１０４Ｃである。ノード１０４Ａは、ノード１０４Ｂがその記憶装置と通信できないことを示す情報を記憶し、その後、新しいデータ・アクセス要求を他のノードに送ることができる。

【００４５】

代替例では、ノード１０４Ｂを動作不能にすることができる。一実施形態では、ノード１０４ＡがＣＭＭ３１０Ａから取得したクラスタ・メンバシップ・データは、そのノードが動作不能であることを示す場合もある。したがって、ＮＤ３１８Ａは、データ・アクセス要求を二次ノードに経路指定することができる。上記のように、障害が存在する状態でもデータ・アクセス要求が正常に完了する。

【００４６】

次に図４に移行すると、本発明の一実施形態によるネットディスク・ドライバの初期設定を示すブロック図が示されている。図４はノード１０４ＡにおけるＮＤ３１８Ａの初期設定を示している。そのクラスタ内の他のネットディスク・ドライバの初期設定は実質的に同じように実行することができる。

【００４７】

一実施形態では、記憶装置にアクセスする前にその記憶装置をオープンする。たとえば、記憶装置を初期設定させるオープン・コマンドを実行することができる。同様に、各ノードの各仮想装置はそれにアクセスする前にオープンすることができる。クライアント３１２Ａは、ＮＤ３１８Ａにコマンドを出力して仮想装置をオープンする。ＮＤ３１８Ａはオープンすべき装置をリストに記憶する。一実施形態のＮＤＤ３１４Ａは、定期的にそのリストに照会して、どの装置を初期設定するかを決定する。代替実施形態のＮＤ３１８Ａは、装置を初期設定する必要があることを示す信号をＮＤＤ３１４Ａに出力することができる。ＮＤＤ３１４Ａは、オープンすべき装置に関するマッピング情報を入手するためにＣＣＤ３１１Ａに照会し、現行メンバシップ情報についてはＣＭＭ３１０Ａに照会する。ＮＤＤ３１４Ａはマッピングおよびメンバシップ情報をＮＤ３１８Ａに送る。Ｎ

Ｄ３１８Ａはマッピングおよびメンバシップ情報を構成ファイルに記憶する。ＮＤ３１８Ａは、構成ファイルに記憶したマッピングおよびメンバシップ・データを使用して、ノードへのデータ・アクセス要求の経路指定を決定する。次にＮＤ３１８Ａは、その装置がオープンされたことをクライアント３１２Ａに通知する。

【００４８】

一実施形態では、各仮想装置に関するマッピング情報としては、仮想装置の名前と、一次ノードと、その一次ノードの記憶装置の名前（すなわち、その記憶装置に対応する装置の名前）と、二次ノードと、その二次ノードの記憶装置の名前とを含む。そのうえ、マッピング情報は、仮想装置の識別番号と、記憶装置のクラスタ特有の名前とを含むこともできる。

【００４９】

そのうえ、ＮＤ３１８Ａは、マッピングおよびメンバシップ・データに関連する再構成番号を記憶する。再構成番号はＣＣＭ３１０Ａから入手される。ＮＤ３１８Ａは再構成番号を使用して、その現行メンバシップ・データが最近の構成に関して最新のものであるかどうかを判定する。

【００５０】

一実施形態では、クラスタの構成が変更されると、ＣＭＭ３１０Ａは、新しいメンバシップ情報をＮＤＤ３１４Ａに通知する。たとえば、ノード障害が検出された場合、ＣＭＭ３１４Ａは、再構成が行われたことをＮＤＤ３１４Ａに通知し、新しいメンバシップ・データをＮＤＤ３１４Ａに送る。ＮＤＤ３１４Ａは新しいメンバシップ情報をＮＤ３１８Ａに送り、そのＮＤ３１８Ａはマッピング情報とともに新しいメンバシップ情報を使用して将来のデータ・アクセス要求を経路指定する。

【００５１】

一実施形態では、ファイルシステムがノードの仮想ディスクを管理する。このファイルシステムはネットディスク・ファイルシステム（ＮＤＦＳ）と呼ぶこともできる。ＮＤＦＳはあるノードが仮想ディスクをオープンしたときにその仮想ディスク用の特殊装置ファイルを作成するように構成されている。この特殊装置

ファイルは、オペレーティング・システム内で仮想ディスクを表すものである。

【００５２】

ＵＮＩＸオペレーティング・システムなどのオペレーティング・システムでは、装置をファイルとして扱うことができる。ある装置に関連するファイル（装置ファイルまたは特殊装置ファイルという）は、通常、オペレーティング・システムのブートアップ・フェーズ中に実行される初期設定プログラムによって作成される。初期設定プログラムは、コンピュータ・システムに接続された物理装置を決定し、その物理装置に対応する装置ファイルを作成する。一実施形態では、ブートアップ中ではなく、最初にアクセスされたときに仮想装置が初期設定される。この状況ならびにその仮想ディスクをノードに物理的に接続できないことは、初期設定中に仮想ディスク用の装置ファイルを作成できないことを意味する。好ましくは仮想ディスクは他の装置のようにアクセス可能なので、ＮＤＦＳは、最初にオープンされたときに仮想装置の装置ファイルを作成するように構成されている。一実施形態では、あるノードがある仮想装置を最初にオープンしたときにのみ、装置ファイルが作成される。その後、その仮想装置をオープンしても、装置ファイルは作成されない。

【００５３】

一実施形態のＮＤＦＳは仮想装置をオープンするコマンドを検出する。これがその仮想装置がオープンされた最初の場合であれば、ＮＤＦＳは作成要求をＮＤ３１８Ａに送る。一実施形態のＮＤＦＳは、ＮＤ３１８Ａへのプライベート・インタフェースを有する。ＮＤ３１８Ａはリストとして作成するためにその仮想装置を記憶する。このリストは、オープンすべき装置を記憶するために使用するのと同じリストである場合もあれば、作成すべき装置用の個別のリストである場合もある。ＮＤＤ３１４Ａが定期的にそのリストに照会してどの装置を作成すべきかを決定する場合もあれば、ある装置を作成する必要があることを示す信号をＮＤ３１８ＡがＮＤＤ３１４Ａに出力する場合もある。ＮＤＤ３１４ＡはＣＣＤ３１１Ａに照会し、オープンすべき装置に関する許可データを入手する。ＮＤＤ３１４ＡはＮＤ３１８Ａに許可データを送り、次にそのＮＤ３１８Ａが許可データをＮＤＦＳに送る。ＮＤＦＳは、ＣＣＤ３１１Ａから受け取った許可データによ

ってその装置に関する装置ファイルを作成することになる。一実施形態では、前述のように通常の装置オープン手順を使用して装置ファイルが作成されたあとで装置がオープンされる。その後、同じノードによって同じ装置をオープンすると、ＮＤＦＳがかかわる必要なしに通常のオープン動作が行われる。したがって、性能上のハンディは装置を最初にオープンしたときにだけ発生する。その装置をオープンするためのその後のコマンドは、他のどの装置のオープンとも同じように実行される。

【００５４】

次に図５に移行すると、本発明の一実施形態によるクラスタ・トランスポート・インタフェースの初期設定を示すブロック図が示されている。図５はノード１０４ＡにおけるＣＴＩ３１６Ａの初期設定を示している。そのクラスタ内の他のクラスタ・トランスポート・インタフェースの初期設定は実質的に同じように実行することができる。

【００５５】

一実施形態では、データ通信リンク１０２によりデータを転送する前に、ＣＴＩＤ３１６Ａは使用可能なリンクにより接続を確立する。初期設定中にＣＴＩＤ３１６Ａは、現行クラスタ・メンバシップを識別するデータを求めてＣＭＭ３１０Ａに照会し、どのリンクがどのノードに接続されるかを識別するデータを求めてＣＣＤ３１１Ａに照会する。一実施形態のＣＣＤ３１１Ａは、リンクの転送プロトコルなど、そのリンクに関する追加情報を記憶する。ＣＴＩＤ３１６Ａは、使用可能なリンクにより接続を確立し、リンク情報およびメンバシップ・データをＣＴＩ３２２Ａに渡す。一実施形態のＣＴＩＤ３１６Ａは使用可能なリンクによりＴＣＰ／ＩＰ接続を確立する。

【００５６】

ＣＴＩ３２２Ａはネットワーク・トランスポート３２８Ａにインタフェースし、ＣＴＩ３２２の他のインスタンスへのデータを交換する。一実施形態のネットワーク・トランスポート３２８ＡはＣＣＭ３２４Ａにインタフェースし、それが１つまたは複数の冗長リンクを管理する。ＣＴＩ３２２Ａは、特定のノード宛てのデータ・アクセス要求を受け取ると、どの接続が要求側ノードを宛先ノードに

接続するかを決定する。ＣＴＩ３２２Ａは、どの接続（複数も可）上で宛先ノードにデータを移送するかを決定する。たとえば、ＣＴＩ３２２Ａがノード１０４Ｂへの３つのリンクによる接続を管理し、それがそのノード宛てのデータ・アクセス要求を受け取る場合、ＣＴＩ３２２Ａは１つの接続を介してすべてのデータを転送する場合もあれば、３つの接続のそれぞれによりデータの一部分を転送する場合もある。

【００５７】

クラスタが再構成されると、ＣＭＭ３１０Ａはその事象をＣＴＩＤ３１６Ａに通知する。ＣＴＩＤ３１６ＡはＣＣＤ３１１Ａから新しいメンバシップ・データを入手し、その新しいメンバシップ・データと新しい構成番号をＣＴＩ３２２Ａに送る。そのうえ、ＣＴＩＤ３１６ＡはＣＣＤ３１１Ａからリンク・データを入手することができ、そのデータをＣＴＩ３２２Ａに送る。ＣＴＩＤ３２２Ａは、再構成が行われたときに接続を変更することができる。たとえば、ＣＴＩＤ３２２Ａは、クラスタ内の新しいノードに対してリンクにより接続を確立する場合もあれば、そのクラスタを離れるノードへの接続を放棄する場合もある。

【００５８】

次に図６に移行すると、本発明の一実施形態による仮想ディスク・システムの動作を示す流れ図が示されている。ステップ６１２では、ネットディスク・ドライバを初期設定する。ネットディスク・ドライバの初期設定については図７に関連して詳述する。ステップ６１４では、クラスタ・トランスポート・ネットワークを初期設定する。クラスタ・トランスポート・インタフェースの初期設定については図８に関連して詳述する。ステップ６１６では、ネットディスク・ドライバがクライアントからデータ・アクセス要求を受け取る。ステップ６１７では、ネットディスク・ドライバは、データ・アクセス要求と、それが正常に完了していない場合にデータ・アクセス要求を再発行するために必要な他のデータを記憶する。

【００５９】

ステップ６１８では、データ・アクセス要求を受け取るネットディスク・ドライバは、宛先装置が要求側ノードに物理的に接続されているかどうかを判定する

。宛先装置が要求側ノードに物理的に接続されている場合、ネットディスク・ドライバはステップ６２０で記憶装置上でデータ・アクセス要求を実行する。あるいは、記憶装置が要求側ノードに物理的に接続されていない場合、ネットディスク・ドライバはステップ６２０でデータ・アクセス要求を送るべきノードを検出する。一実施形態のネットディスク・ドライバは、各記憶装置ごとに一次および二次ノードを識別するマッピング情報を記憶する。特定の一実施形態では、ネットディスク・ドライバは、メンバシップ・データおよび／または前の不成功に終わったデータ・アクセス要求に基づいて、一次または二次ノードを選択する。ステップ６２４では、ネットディスク・ドライバは、クラスタ・トランスポート・インタフェースを介して選択した宛先ノードにデータ・アクセス要求を送る。

【００６０】

ステップ６２６では、クラスタ・トランスポート・インタフェースは、ネットディスク・ドライバにより宛先ノードにデータを転送するために１つまたは複数の接続を選択する。ステップ６２８では、クラスタ・トランスポート・インタフェースは、選択した接続（複数も可）を介して宛先ノードにデータ・アクセス要求を送る。ステップ６３０では、宛先ノード側のクラスタ・トランスポート・インタフェースは、データ・アクセス要求を受け取って宛先クライアントを決定するが、その宛先クライアントはこの例ではネットディスク・ドライバ、またはより詳細にはネットディスク・マスタである。ステップ６３２では、ネットディスク・マスタがデータ・アクセス要求を受け取り、宛先記憶装置にアクセスする。ステップ６３４では、宛先ノードのクラスタ・トランスポート・インタフェースが肯定応答または否定応答信号を要求側ノードに返す。データ・アクセス要求が読取り要求である場合、要求されたデータも要求側ノードに返されるであろう。

【００６１】

データ・アクセス要求の転送と並行して、ステップ６３８では、要求側ノードが宛先ノードからの状況信号を待つ。この状況信号は肯定応答または否定応答信号の形を取ることができる。ステップ６４０では、肯定応答が受け取られたかどうかを判定する。肯定応答信号が受け取られた場合、動作はステップ６１６に継続する。あるいは、否定応答信号が受け取られた場合、ステップ６４２では、デ

ータ・アクセス要求を送るための代替ノードを選択し、動作はステップ６２４に継続する。

【００６２】

次に図７に移行すると、本発明の一実施形態によるネットディスク・ドライバの初期設定を示す流れ図が示されている。ステップ７１２では、ネットディスク・デーモンは、オープンすべき装置を求めてネットディスク・ドライバに照会する。判断ステップ７１４では、オープンする必要がある装置があるかどうかを判定する。どの装置もオープンする必要がない場合、実行はステップ７１２に継続する。あるいは、ネットディスク・デーモンがオープンすべき装置を検出した場合、ステップ７１６でネットディスク・デーモンがマッピング・データを求めてクラスタ構成データベースに照会する。このマッピング・データは仮想装置にマッピングされたノード／ディスク対を識別することができる。ステップ７１８では、ネットディスク・デーモンはメンバシップ・データを求めてクラスタ・メンバシップ・モニタに照会する。

【００６３】

ステップ７２０では、ネットディスク・デーモンはマッピングおよびメンバシップ・データをネットディスク・ドライバに送る。ステップ７２２では、ネットディスク・ドライバは、その装置に関するマッピング情報を更新し、そのメンバシップ情報を更新し、再構成番号を記録する。ステップ７２４では、ネットディスク・ドライバは、要求された装置がオープンされていることをクライアントに通知する。

【００６４】

次に図８に移行すると、本発明の一実施形態によるクラスタ・トランスポート・インタフェースの初期設定を示す流れ図が示されている。ステップ８１２では、クラスタ・トランスポート・インタフェース・デーモンは構成変更の表示を受け取る。あるいは、クラスタ・トランスポート・デーモンは、システム初期設定の表示を受け取る場合もある。ステップ８１４では、クラスタ・トランスポート・インタフェース・デーモンは、リンク情報を求めてクラスタ構成データベースに照会する。一実施形態では、リンク情報は、あるクラスタ内のノード間のリン

クの数と、どのリンクがどのノードに結合されているかということ、そのリンクが使用するプロトコルなどの情報とを含むことができる。ステップ８１６では、クラスタ・トランスポート・インタフェース・デーモンはメンバシップ情報を求めてクラスタ・メンバシップ・モニタに照会する。

【００６５】

ステップ８１８では、クラスタ・トランスポート・インタフェースがリンクにより接続を確立する。ステップ８２０では、クラスタ・トランスポート・インタフェース・デーモンがクラスタ・トランスポート・インタフェースにリンクおよびメンバシップ情報を送る。その場合、クラスタ・トランスポート・インタフェースはデータ・アクセス要求またはその他のメッセージを受け入れる準備が整っている。

【００６６】

次に図９に移行すると、本発明の一実施形態によるクラスタ・トランスポート・インタフェースのブロック図が示されている。クラスタ・トランスポート・インタフェースはデータ・トランスポート・システムの一例である。図９は、クラスタ・トランスポート・インタフェースの３つのインスタンス（３２２Ａ～３２２Ｃ）と、３つのＴＣＰ／ＩＰインタフェース（９１２Ａ～９１２Ｃ）と、８つのクラスタ接続モニタ（９１４Ａ～９１４Ｈ）とを含む。ＣＴＩ３２２は、ノード間でメッセージを受け渡すための機能を備えた分散ソフトウェア・プログラムである。そのメッセージとしては、制御メッセージとデータ・ブロックを含むことができる。

【００６７】

クラスタ・トランスポート・インタフェース３２２のインスタンスは、クライアント・プログラム間でデータを受け渡す。たとえば、ＣＴＩ３２２Ａは、ＣＴＩ３２２Ａにとってクライアントであるネットディスク・ドライバからメッセージを受け取ることができる。一実施形態のメッセージは、その宛先ノードと、そのノードのディスク装置を指定するものである。ＣＴＩ３２２Ａは、どのリンクが宛先ノードに接続されるかを決定し、そのリンクのうちの１つによりメッセージを送る。宛先ノード側のクラスタ・トランスポート・インタフェースは、デー

タ・アクセス要求を受け取り、宛先クライアントを決定し、宛先クライアントにデータを送る。たとえば、ＣＴＩ３２２Ａは、ノード１０４Ａ内のネットディスク・ドライバからノード１０４Ｂ内のネットディスク・ドライバにデータ・アクセス要求を経路指定することができる。ＣＴＩ３２２Ｂは、データ・アクセス要求を受け取り、宛先クライアントを決定し、ノード１０４Ｂ内のネットディスク・ドライバにデータ・アクセス要求を送る。クライアントの見地からすると、ＣＴＩ３２２Ａは宛先ノードへの仮想リンクの１つとして現れる。

【００６８】

図示の実施形態では、ＣＴＩ３２２は、他のノードにデータを転送するためにＴＣＰ／ＩＰを使用する。ＣＴＩＤ３１６Ａは、初期設定中に各リンクによりＴＣＰ／ＩＰ接続を自動的に確立する。ＣＴＩ３２２は、ＣＣＭ９１４の適切なインスタンスにメッセージを転送するＴＣＰ／ＩＰ９１２Ａにメッセージを送る。しかし、ＣＴＩ３２２Ａは、特定のデータ転送プロトコルに依存していない。ＴＣＰ／ＩＰ９１２および／またはＣＣＭ９１４を変更することにより、ＣＴＩ３２２はどのようなデータ・トランスポート・インタフェースまたは転送プロトコルにもインタフェースすることができる。

【００６９】

一実施形態のＣＴＩ３２２Ａは、他のノードから受け取ったメッセージおよびデータを記憶するためのメモリを割り振り、クライアントがもはやそのデータを必要としなくなったときにそのメモリの割振りを解除する。一実施形態のＣＴＩ３２２は、コールバック機能を使用して、そのデータを受け取ったことをクライアントに示す。たとえば、ＣＴＩ３２２Ａはノード１０４Ｂに読取り要求を送ることができる。ＣＴＩ３２２Ａは、要求されたデータを受け取ると、要求側クライアントへのコールバック機能を使用して、そのデータが使用可能であることを示す。

【００７０】

クラスタ接続モニタ（ＣＣＭ）９１４は、２つまたはそれ以上の物理リンクを１つの論理リンクとして管理する。図示の実施形態では、ＣＣＭ９１４の一対のインスタンスが２つのリンクを管理する。代替実施形態では、ＣＣＭ９１４の一

対のインスタンスがそれ以上のリンクを管理することができる。複数対の物理リンクはそのクラスタ内のあるノードを他のノードに接続する。たとえば、リンク９１６Ａはノード１０４Ａをノード１０４Ｂに結合し、リンク９１６Ｂはノード１０４Ａをノード１０４Ｃに結合する。一実施形態では、ＣＭＭ９１４によってリンクが冗長リンクとして扱われる。データは一方のリンクの障害が検出されるまでそのリンク上で転送され、その後、データはもう一方のリンク上で転送される。

【００７１】

ＣＣＭ９１４は、どのリンクが動作可能であるかを決定し、両方の物理リンクにより、ときにはハートビート・メッセージと呼ばれるメッセージを交換することで障害を検出する。たとえば、ＣＣＭ９１４ＡとＣＣＭ９１４Ｅは、ハートビート・メッセージを交換して、物理リンク９１６Ａが動作可能であるかどうかを判定する。ＣＣＭ９１４の２つのインスタンスは物理リンクのうちの一方を一次リンクとして選択する。一次リンクが故障した場合、ＣＣＭ９１６はその障害を検出し、もう一方の物理リンク上でデータの転送を開始する。特定の一実施形態では、ＣＣＭ９１６は、物理リンクを越えて不信頼データ・プロトコル（ＵＤＰ）メッセージを交換して、そのリンクが動作可能であるかどうかを判定する。

【００７２】

ＣＴＩ３２２の見地からすると、ＣＣＭ９１４によって管理される各対の物理リンクは１つの論理リンクとして現れる。したがって、ＣＴＩ３２２Ａによって転送されるデータは、ＣＴＩ３２２Ａにとってトランスペアレントな２つの物理リンクのうちの一方で転送することができる。

【００７３】

図示の実施形態では、３つの論理リンク（９１６Ｂ～９１６Ｄ）がノード１０４Ａをノード１０４Ｃに接続する。ＣＴＩ３２２Ａは、３つのリンクのうちのどのリンク上でデータを転送するかを決定する。一実施形態のＣＴＩ３２２Ａは１つの論理リンク上ですべてのデータを転送することができる。代替実施形態のＣＴＩ３２２Ａは各論理リンク上でデータの一部分を転送することができる。上記のように、どの論理リンク上またはいくつの論理リンク上でデータを転送するか

はクライアントにとってトランスペアレントなものである。

【００７４】

次に図１０に移行すると、本発明の一実施形態による装置許可を示す図が示されている。許可データはディレクトリのリストに関連して示されている。同様のリストは、生の仮想ディスク装置をリストするディレクトリで「ｌｓ－ｌ」コマンドを実行することによって入手することができる。装置許可は装置そのものに関するものであって、その装置上のファイルまたはディレクトリに関するものではないことに留意されたい。生の装置（すなわち、その上にファイルシステムまたはファイルが一切置かれていない装置）は許可目的のためのファイルとして扱われる。

【００７５】

フィールド１０１２は１０個の文字を含む。第１の文字は、ディレクトリを識別する「ｄ」または装置を識別する「－」のいずれかである。次の９つの文字は、３文字ずつ３つのグループである。各グループは、所有者、グループ、その他のための許可モードをそれぞれ表している。許可モードとしては、読取り（ｒ）、書込み（ｗ）、実行（ｘ）を含む。各グループ内の１つの文字は各許可モードを表す。許可モードを表す英字が存在する場合、関連ユーザはその許可を得ている。あるいは、「－」が存在する場合、関連ユーザはその許可を得ていない。たとえば、あるユーザが「ｒｗｘ」という許可を得ている場合、そのユーザは、読取り、書込み、実行の各許可を得ていることになる。あるいは、ユーザが「ｒ－－」という許可を得ている場合、そのユーザは読取り許可を得ているが、書込みまたは実行の各許可を得ていないことになる。第１のグループの３つの文字はその装置の所有者ための許可を表している。第２のグループの３つの文字はその装置に関連するグループのための許可を表している。最後のグループの３つの文字は他のユーザのための許可を表している。所有者とグループについては以下に詳述する。たとえば、フィールド１０１２内の許可が「ｄｒｗｘ－－ｘ－ｘ－」である場合、そのフィールドはディレクトリを表し、所有者が読取り、書込み、実行の各許可を得ており、グループとその他が実行許可のみを得ていることになる。

【００７６】

フィールド１０１６はその項目の所有者を識別する。所有者はその装置を作成したユーザである。フィールド１０１８は関連ユーザのグループを識別する。グループはオペレーティング・システム内で定義される。フィールド１０１８は定義済みグループの１つを装置に関連付けるものである。他のユーザは所有者ではなく、選択したグループ内にも含まれない。前述のように、所有者、グループ、その他のユーザのために、それぞれ異なる許可を定義することができる。

【００７７】

フィールド１０２２はその装置の最後の変更の日付と時刻を識別する。最後の変更が現行暦年の範囲内である場合、月、日、時刻が指定される。あるいは、最後の変更が現行暦年の範囲内ではない場合、月、日、年が指定される。フィールド１０２４は装置の名前を識別する。

【００７８】

クラスタのノード間で一貫した許可データを保証するため、許可データは高可用性データベースに記憶することができる。一実施形態では、あるクラスタ内の複数のノードがある装置の表現を有する。障害が存在する状態でもノード間で一貫した許可データを維持するため、許可データはクラスタ構成データベース（ＣＣＤ）に記憶される。

【００７９】

一実施形態では、あるノードが最初に仮想装置をオープンすると、その装置のための許可データがＣＣＤから読み取られ、その許可データによって装置ファイルが作成される。一実施形態の装置ファイルは、仮想装置があるノードによって最初にオープンされたときにのみ作成される。一実施形態では、各ノード上で動作するファイルシステムは、その装置の許可データを求めてＣＣＤに照会するデーモンを含む。このデーモンは許可データをファイルシステムに返し、そのファイルシステムがその許可によって特殊装置ファイルを作成する。ＣＣＤはそのクラスタのどのノードでも照会することができ、障害が存在する状態でも一貫した情報を返すので、すべてのノードが一貫した許可データを有することになる。

【００８０】

次に図１１に移行すると、本発明の一実施形態による一貫した許可データの記憶およびアクセスを示す流れ図が示されている。ステップ１１１２では、許可データを高可用性データベースに記憶する。一実施形態の許可データは、装置許可と、装置の所有者と、装置に関連するグループとを含む。ステップ１１１４では、第１のノードは第１のノードの装置をオープンし、高可用性データベースからの許可データにアクセスする。ステップ１１１５では、そのノードは、許可データを含む、その装置に関する特殊装置ファイルをオープンする。ステップ１１１６では、第２のノードは、第２のノード上にあって同じ物理装置に対応する装置をオープンし、許可データにアクセスする。ステップ１１１７では、そのノードは、第２のノードの許可データを含む、その装置に関する特殊装置ファイルをオープンする。高可用性データベースは一貫したデータを返すので、ノードは一貫した許可データを受け取る。

【００８１】

次に図１２に移行すると、本発明の一実施形態による構成マッピングの更新を示す流れ図が示されている。ステップ１２１２では、更新が保留中であるという表示をノードに提供する。ステップ１２１４では、ノードは記憶装置へのデータ・アクセス要求を中断する。ステップ１２１６では、ノードは未処理のデータ・アクセス要求が完了するのを待つ。ステップ１２１８では、ノードは記憶装置に対する仮想ディスクのマッピングの内部表現を無効にする。ステップ１２２０では、ノードは、内部マッピング表現が無効になり、データ・アクセス要求が中断され、未処理のデータ・アクセス要求が完了したことを示す肯定応答信号を出力する。ステップ１２２２では、システムはすべてのアクティブ・ノードからの肯定応答信号を待つ。ステップ１２２４では、システムはそのマッピングを更新する。ステップ１２２６では、システムは、その更新が完了したという表示を出力する。ステップ１２２８では、ノードはマッピングの更新済みバージョンを要求する。ステップ１２３０では、ノードは、記憶装置へのデータ・アクセス要求の送信を再開する。

【００８２】

一実施形態での更新手順は、クラスタ構成データベース（ＣＣＤ）によって調

整される。エラーを防止するため、マッピングはすべてのノード間で一貫して更新しなければならない。ＣＣＤは、保留中の更新をノードに通知し、その更新が完了したことを同期コマンドによりノードに通知する。一実施形態の同期コマンドは、ＣＣＤ内の行が変更されるたびに必ず実行される。ＣＣＤ内の行の変更中に実行すべきコマンドは、ＣＣＤに記憶されたデータに関連するフォーマット行で指定することができる。同期コマンドは、そのクラスタのすべてのノード上で並行して実行することができる。一実施形態のネットディスク同期コマンドは、ネットディスク・マッピングが変更されたときに実行される。ネットディスク同期コマンドの異なる呼出しは、変更のタイプに応じて実行することができる。ＣＣＤは、マッピングを変更する前に第１の同期コマンドを出力する。第２の同期コマンドは、データベースを更新したあとで実行することができる。

【００８３】

一実施形態では、すべてのノードから肯定応答信号を受け取ったわけではない場合、クラスタはその更新を中断し、取消し信号を出力することになる。一実施形態では、取消し信号によりノードが内部マッピング表現を再確認し、動作を続行する。

【００８４】

上記のように、クラスタの構成は、データを失わずにクラスタが動作している間に変更することができる。システム内のデータ・アクセス要求は遅延する可能性があるが、エラーなしで続行される。上記の再構成手順では、データを失わずに接続を再構成することもできる。たとえば、記憶装置はあるノードから切断して、他のノードに再接続することができる。物理的な再構成はステップ１２２２と１２２４の間で行うことができる。さらに、この再構成は、遅延を除き、クライアントにとってトランスペアレントなものである。上記の再構成の他の応用例としては、動作中にボリューム・マネージャのマッピング（または管理）を変更することがある。

【００８５】

上記の開示内容を完全に理解すると、当業者には多数の変形形態および修正形態が明らかになるだろう。特許請求の範囲はこのような変形形態および修正形態

をすべて包含するものと解釈することを意図するものである。

【図面の簡単な説明】

【図１】

本発明の一実施形態によるクラスタ構成のブロック図である。

【図２】

本発明の一実施形態による代替クラスタ構成のブロック図である。

【図３】

本発明の一実施形態によるクラスタの２つのノード上で動作する仮想ディスク・システムのブロック図である。

【図４】

本発明の一実施形態によるネットディスク・ドライバの初期設定を示すブロック図である。

【図５】

本発明の一実施形態によるクラスタ・トランスポート・インタフェースの初期設定を示すブロック図である。

【図６ａ】

本発明の一実施形態による仮想ディスク・システムの動作を示す流れ図である。

【図６ｂ】

本発明の一実施形態による仮想ディスク・システムの動作を示す流れ図である。

【図７】

本発明の一実施形態によるネットディスク・ドライバの開始を示す流れ図である。

【図８】

本発明の一実施形態によるクラスタ・トランスポート・インタフェースの開始を示す流れ図である。

【図９】

本発明の一実施形態によるクラスタ・トランスポート・インタフェースのブロ

ック図である。

【図１０】

本発明の一実施形態による許可データを示す図である。

【図１１】

本発明の一実施形態による一貫した許可データの記憶およびアクセスを示す流れ図である。

【図１２】

本発明の一実施形態による構成マッピングの更新を示す流れ図である。

【図１】

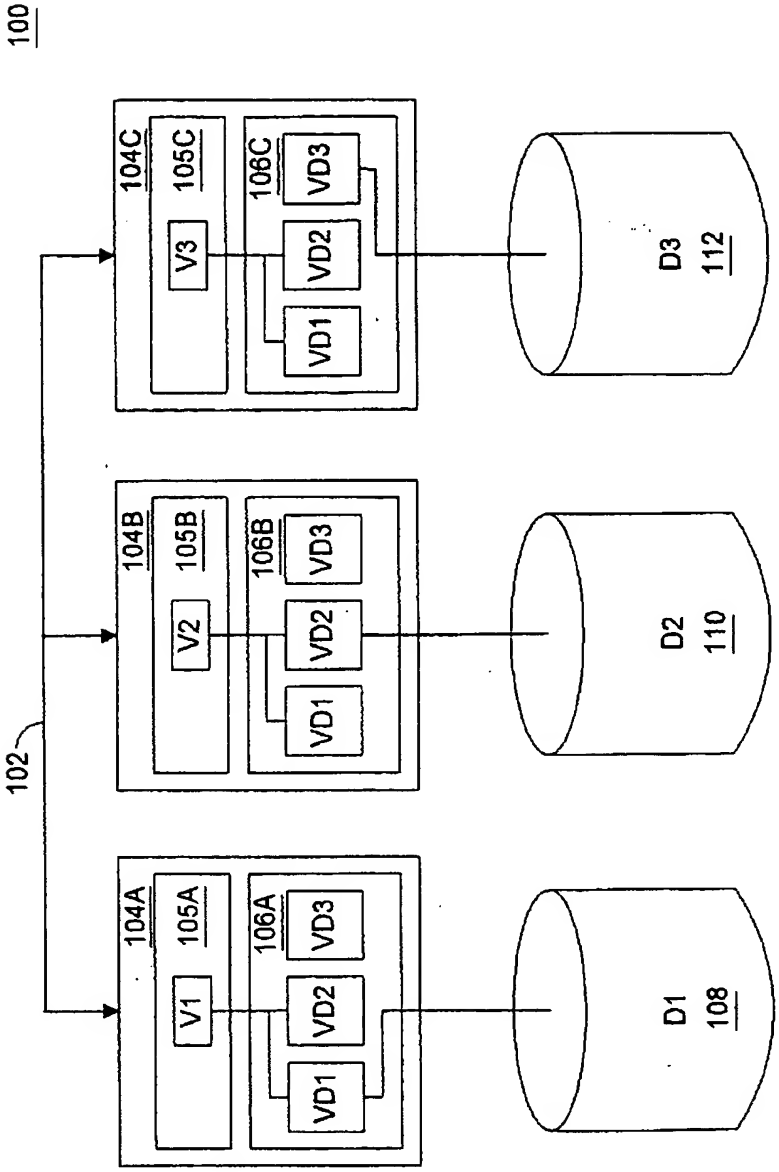

100
102
104A
105A
V1
106A
VD3
VD2
VD1
D1
108
104B
105B
V2
106B
VD3
VD2
VD1
D2
110
104C
105C
V3
106C
VD3
VD2
VD1
D3
112

FIG. 1

【図２】

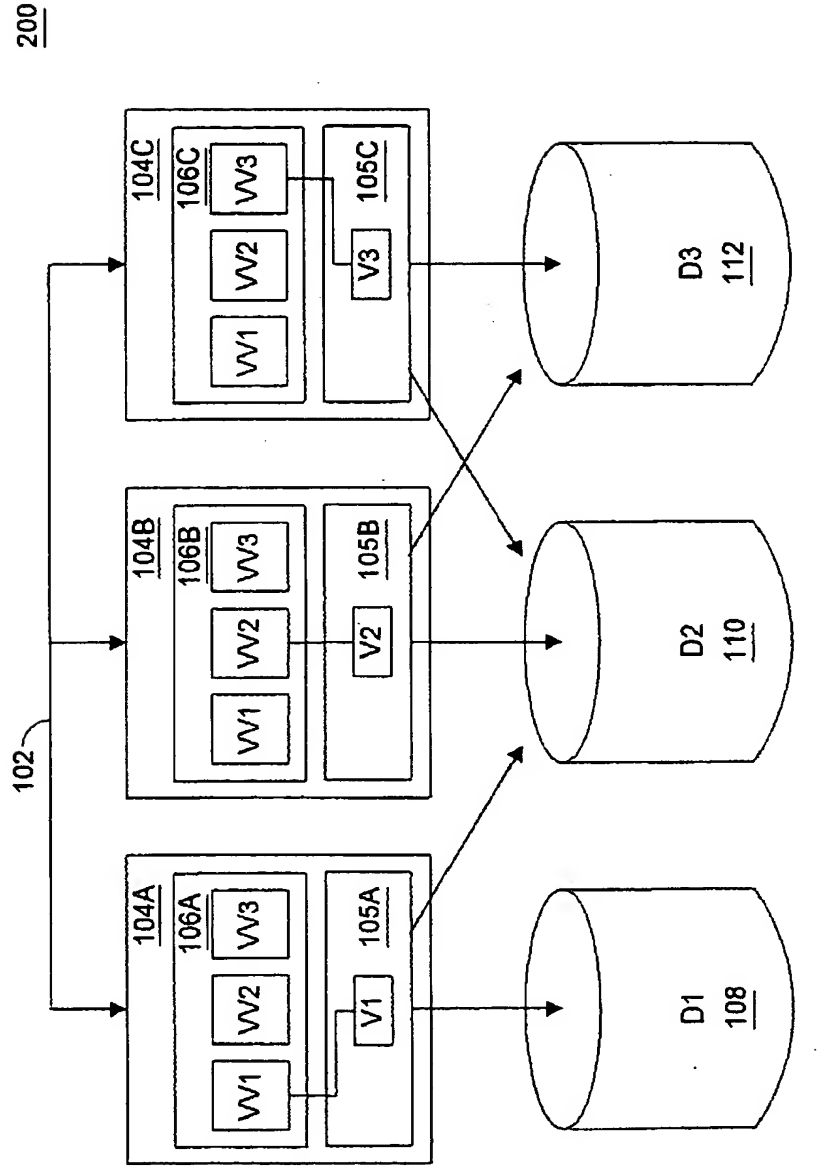
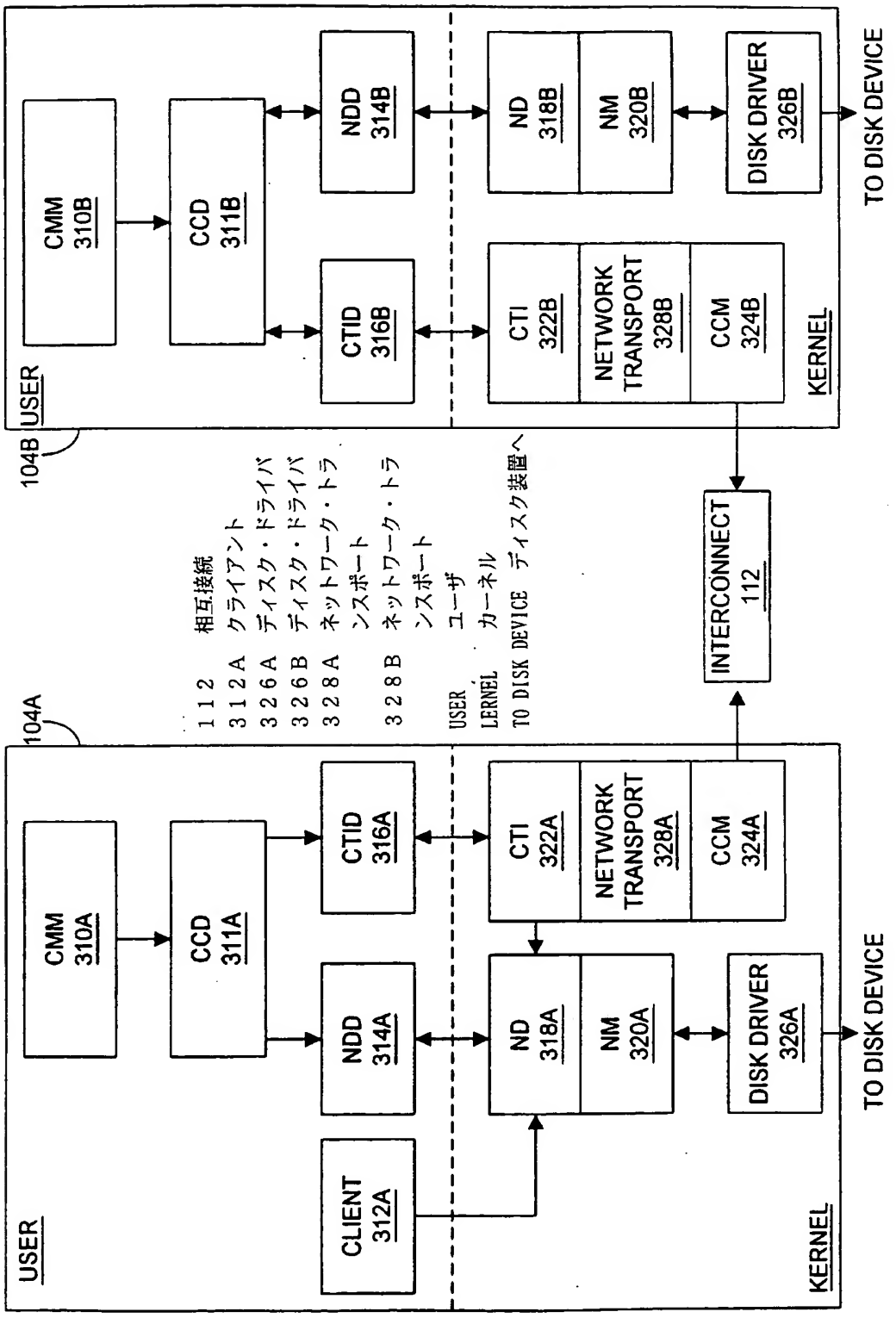
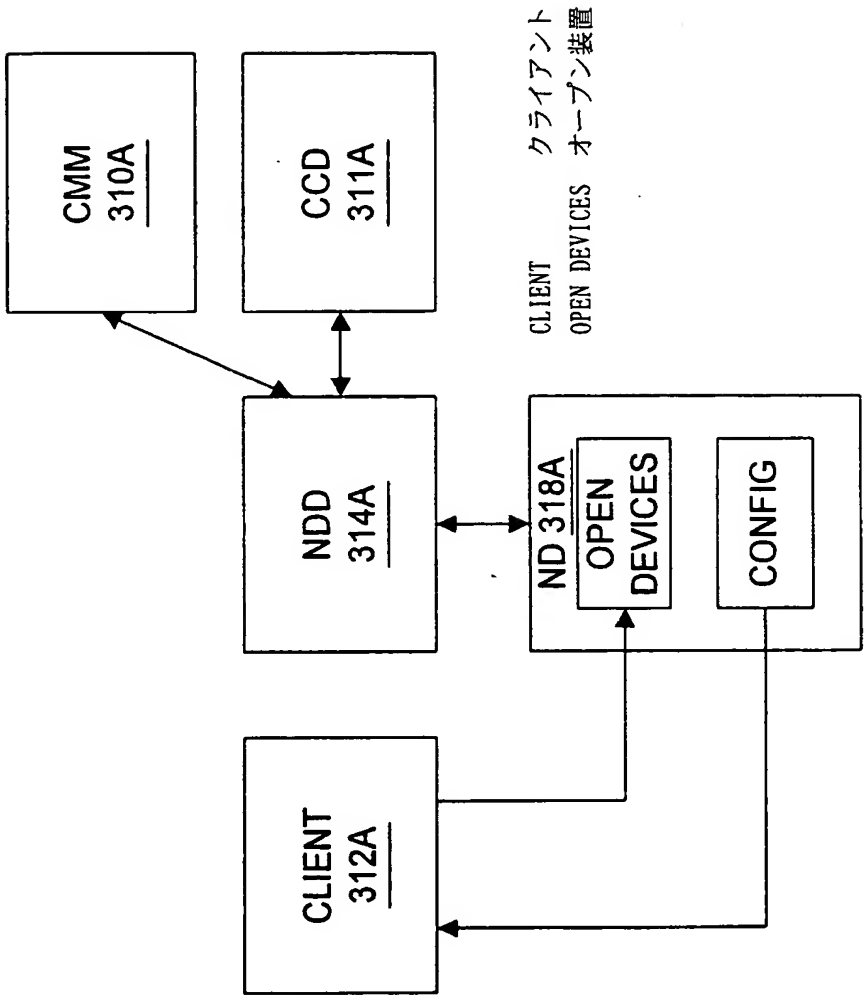

200
102
104A
106A
VV1
VV2
VV3
105A
V1
104B
106B
VV1
VV2
VV3
105B
V2
104C
106C
VV1
VV2
VV3
105C
V3
D1
108
D2
110
D3
112

FIG. 2

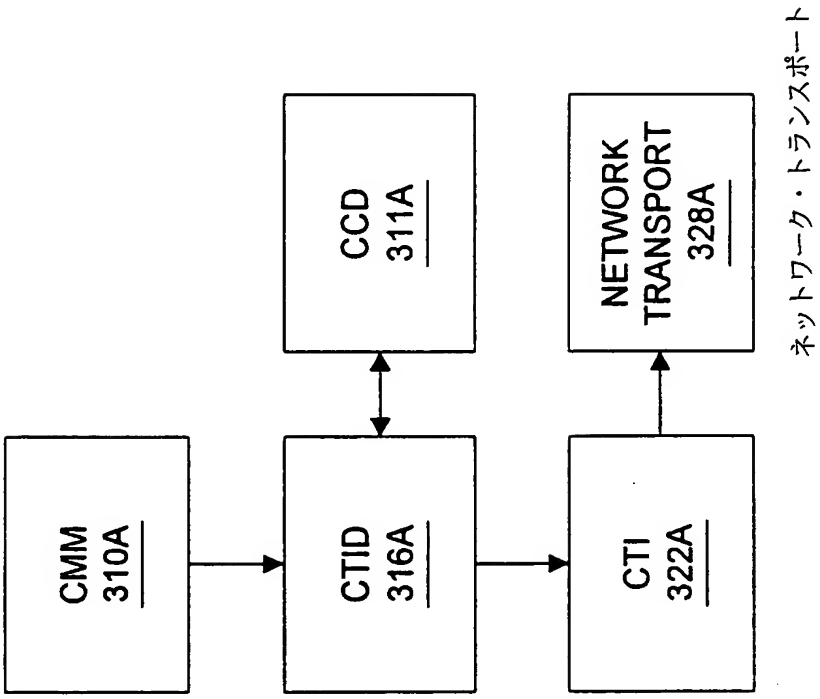
【図３】


USER
KERNEL
104A
104B
CLIENT 312A
CMM 310A
CCD 311A
NDD 314A
CTID 316A
ND 318A
NM 320A
CTI 322A
NETWORK TRANSPORT 328A
CCM 324A
DISK DRIVER 326A
TO DISK DEVICE
CMM 310B
CCD 311B
NDD 314B
CTID 316B
ND 318B
NM 320B
CTI 322B
NETWORK TRANSPORT 328B
CCM 324B
DISK DRIVER 326B
TO DISK DEVICE
INTERCONNECT 112
112 相互接続
312A クライアント
326A ディスク・ドライバ
326B ディスク・ドライバ
328A ネットワーク・トランスポート
328B ネットワーク・トランスポート
USER ユーザ
KERNEL カーネル
TO DISK DEVICE ディスク装置へ

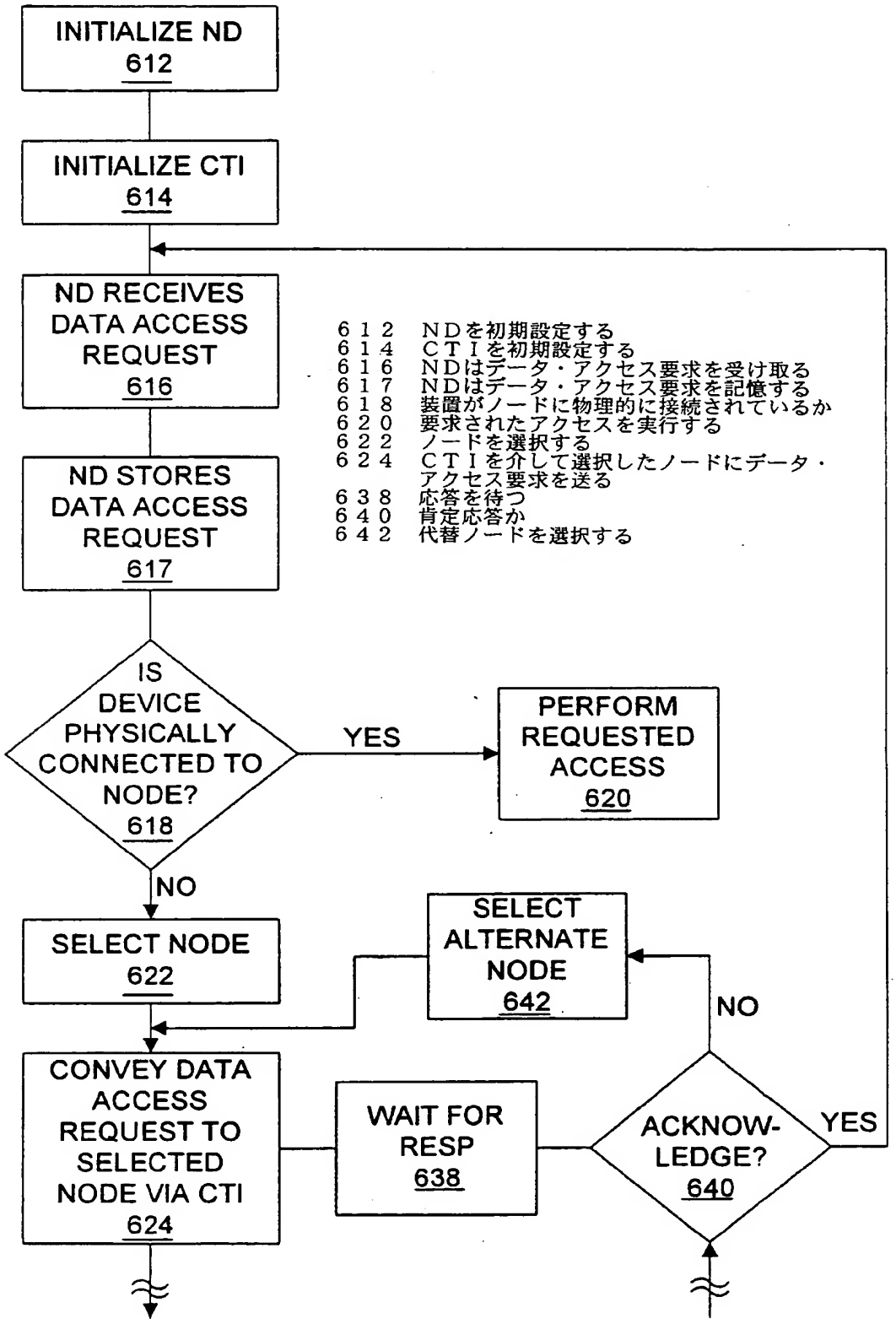
【図４】


CMM
310A
CCD
311A
CLIENT クライアント
OPEN DEVICES オープン装置
NDD
314A
ND 318A
OPEN
DEVICES
CONFIG
CLIENT
312A

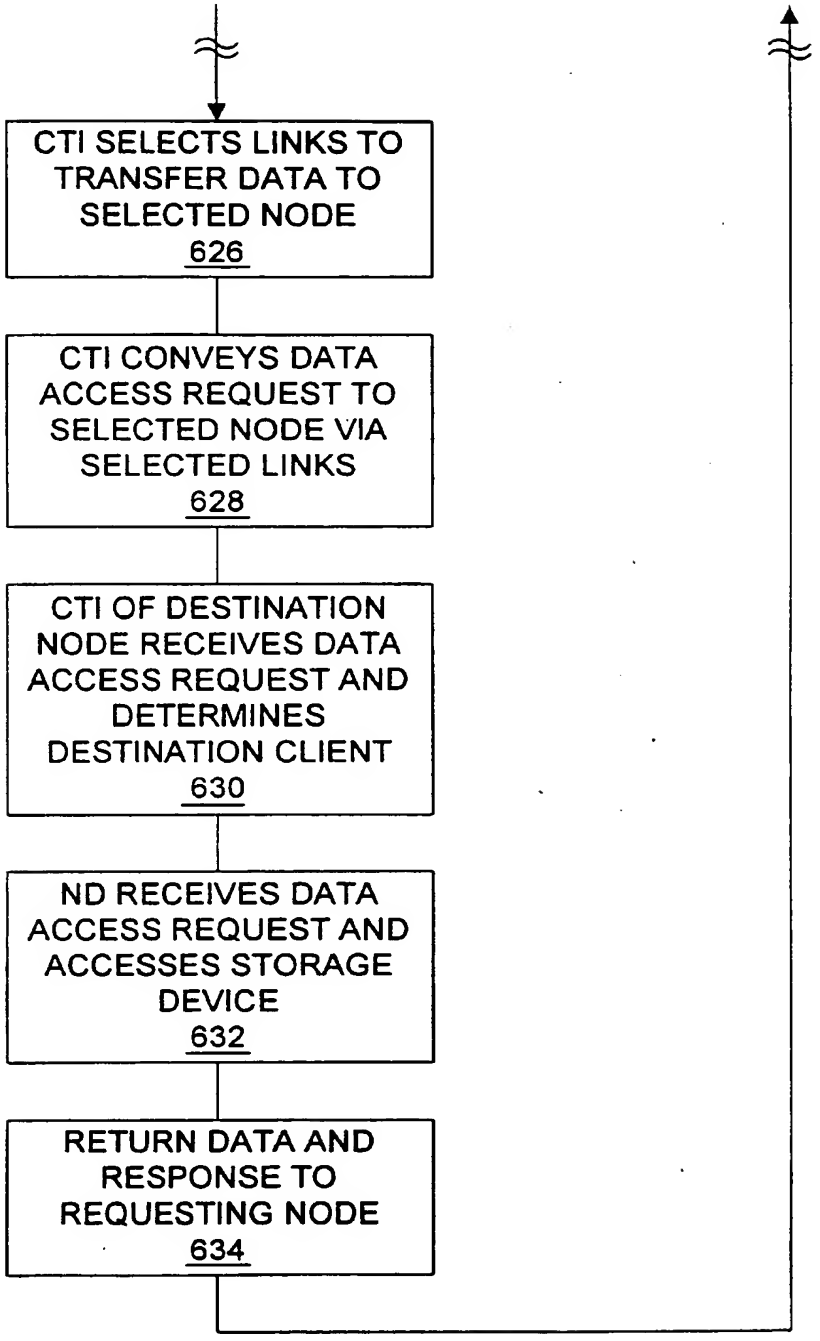
【図５】


CMM
310A
CTID
316A
CTI
322A
CCD
311A
NETWORK
TRANSPORT
328A
ネットワーク・トランスポート

【図６ａ】


INITIALIZE ND
612
INITIALIZE CTI
614
ND RECEIVES DATA ACCESS REQUEST
616
ND STORES DATA ACCESS REQUEST
617
IS DEVICE PHYSICALLY CONNECTED TO NODE?
618
YES
PERFORM REQUESTED ACCESS
620
NO
SELECT NODE
622
SELECT ALTERNATE NODE
642
NO
CONVEY DATA ACCESS REQUEST TO SELECTED NODE VIA CTI
624
WAIT FOR RESP
638
ACKNOW-LEDGE?
640
YES
612 ＮＤを初期設定する
614 ＣＴＩを初期設定する
616 ＮＤはデータ・アクセス要求を受け取る
617 ＮＤはデータ・アクセス要求を記憶する
618 装置がノードに物理的に接続されているか
620 要求されたアクセスを実行する
622 ノードを選択する
624 ＣＴＩを介して選択したノードにデータ・アクセス要求を送る
638 応答を待つ
640 肯定応答か
642 代替ノードを選択する

【図６ｂ】

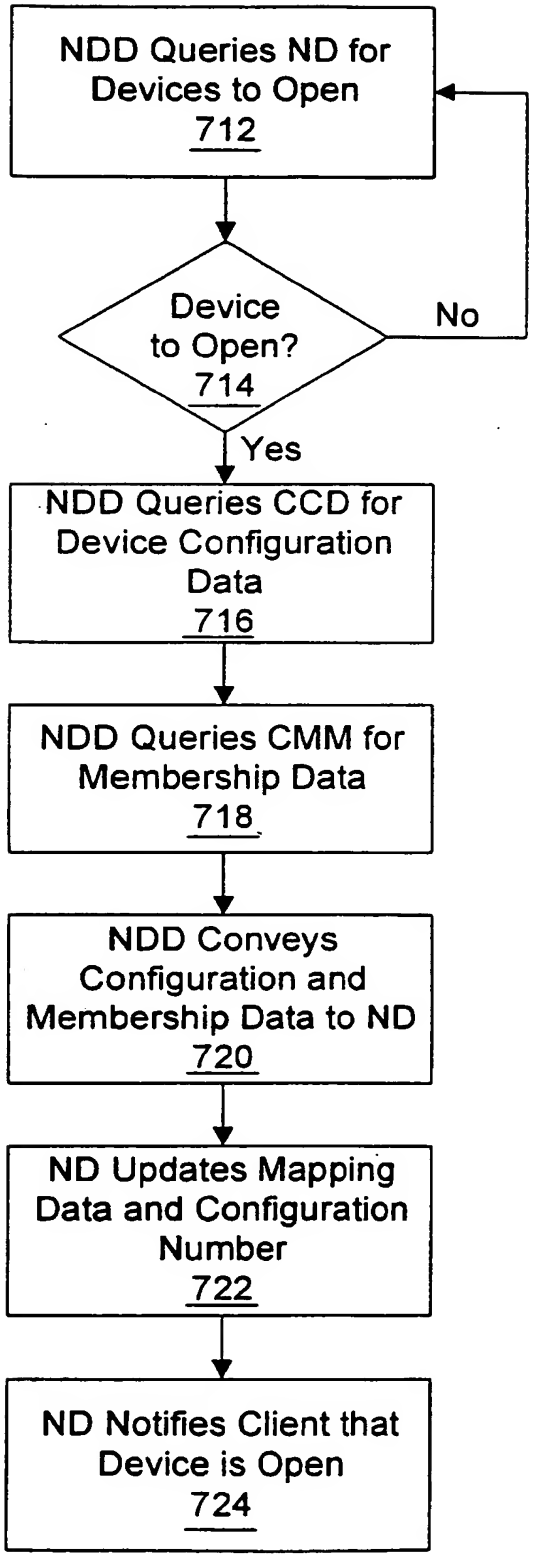


６２６　ＣＴＩは選択したノードにデータを転送するためのリンクを選択する
６２８　ＣＴＩは選択したリンクを介して選択したノードにデータ・アクセス要求を送る
６３０　宛先ノードのＣＴＩはデータ・アクセス要求を受け取り、宛先クライアントを決定する
６３２　ＮＤはデータ・アクセス要求を受け取り、記憶装置にアクセスする
６３４　要求側ノードにデータと応答を返す

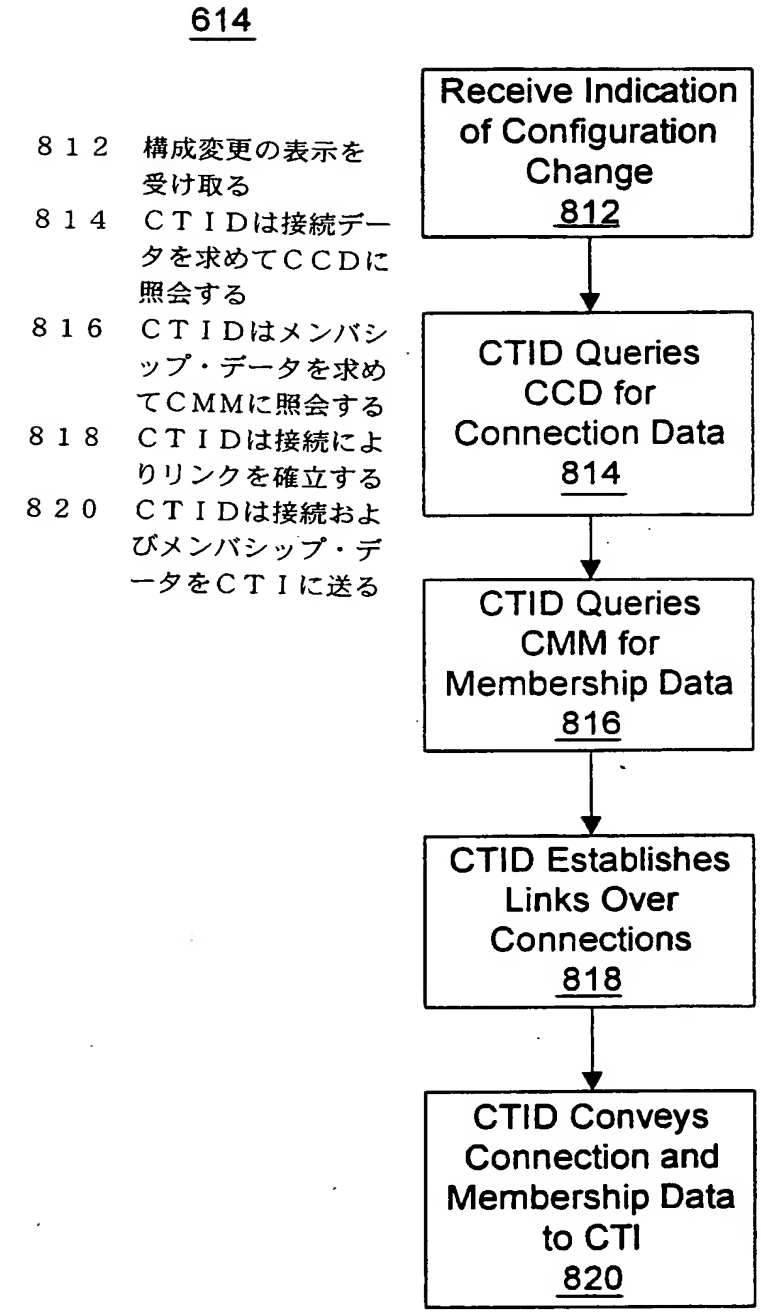
【図７】

612

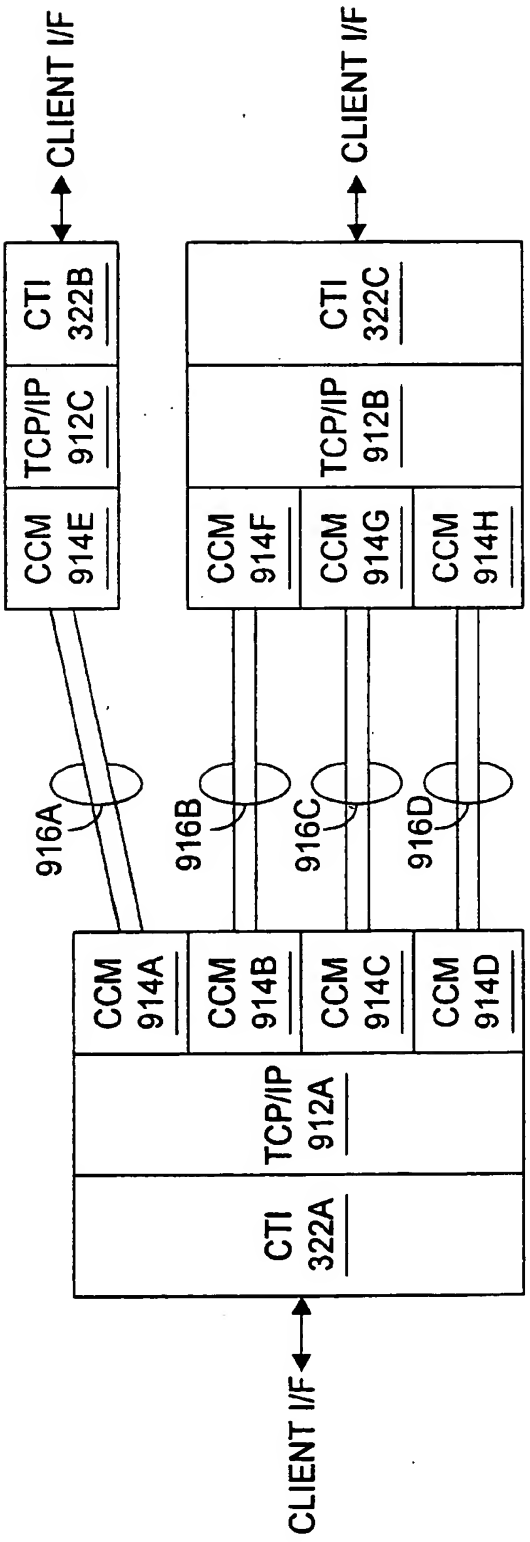
７１２　ＮＤＤはオープンすべき装置を求めてＮＤに照会する
７１４　オープンすべき装置があるか
７１６　ＮＤＤは装置構成データを求めてＣＣＤに照会する
７１８　ＮＤＤはメンバシップ・データを求めてＣＭＭに照会する
７２０　ＮＤＤは構成およびメンバシップ・データをＮＤに送る
７２２　ＮＤはマッピング・データおよび構成番号を更新する
７２４　ＮＤは装置がオープンされていることをクライアントに通知する



【図８】


614
８１２ 構成変更の表示を受け取る
８１４ ＣＴＩＤは接続データを求めてＣＣＤに照会する
８１６ ＣＴＩＤはメンバシップ・データを求めてＣＭＭに照会する
８１８ ＣＴＩＤは接続によりリンクを確立する
８２０ ＣＴＩＤは接続およびメンバシップ・データをＣＴＩに送る
Receive Indication of Configuration Change 812
CTID Queries CCD for Connection Data 814
CTID Queries CMM for Membership Data 816
CTID Establishes Links Over Connections 818
CTID Conveys Connection and Membership Data to CTI 820

【図９】



CLIENT クライアント

【図１０】

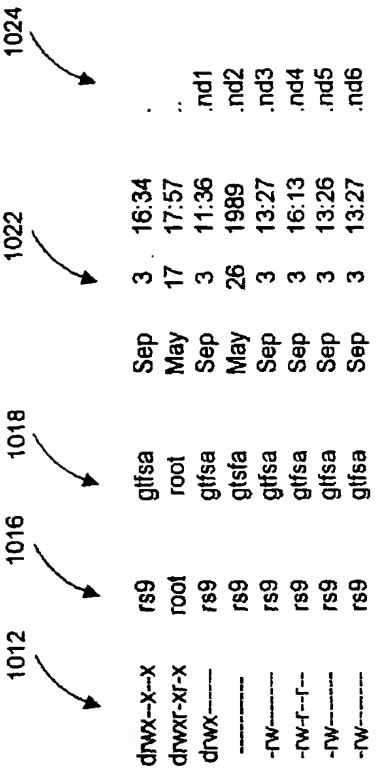
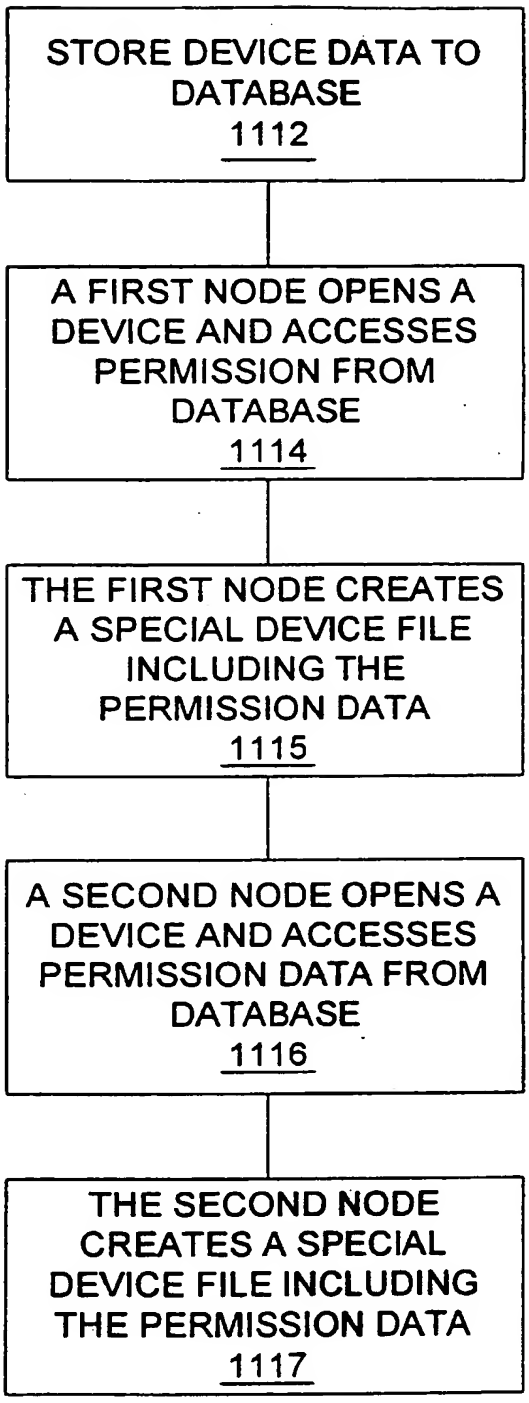

1024
1022
1018
1016
1012
drwx--x--x rs9 gtfsa Sep 3 16:34 .
drwxr-xr-x root root May 17 17:57 ..
drwx------ rs9 gtfsa Sep 3 11:36 .nd1
---------- rs9 gtsfa May 26 1989 .nd2
-rw------- rs9 gtfsa Sep 3 13:27 .nd3
-rw-r--r-- rs9 gtfsa Sep 3 16:13 .nd4
-rw------- rs9 gtfsa Sep 3 13:26 .nd5
-rw------- rs9 gtfsa Sep 3 13:27 .nd6

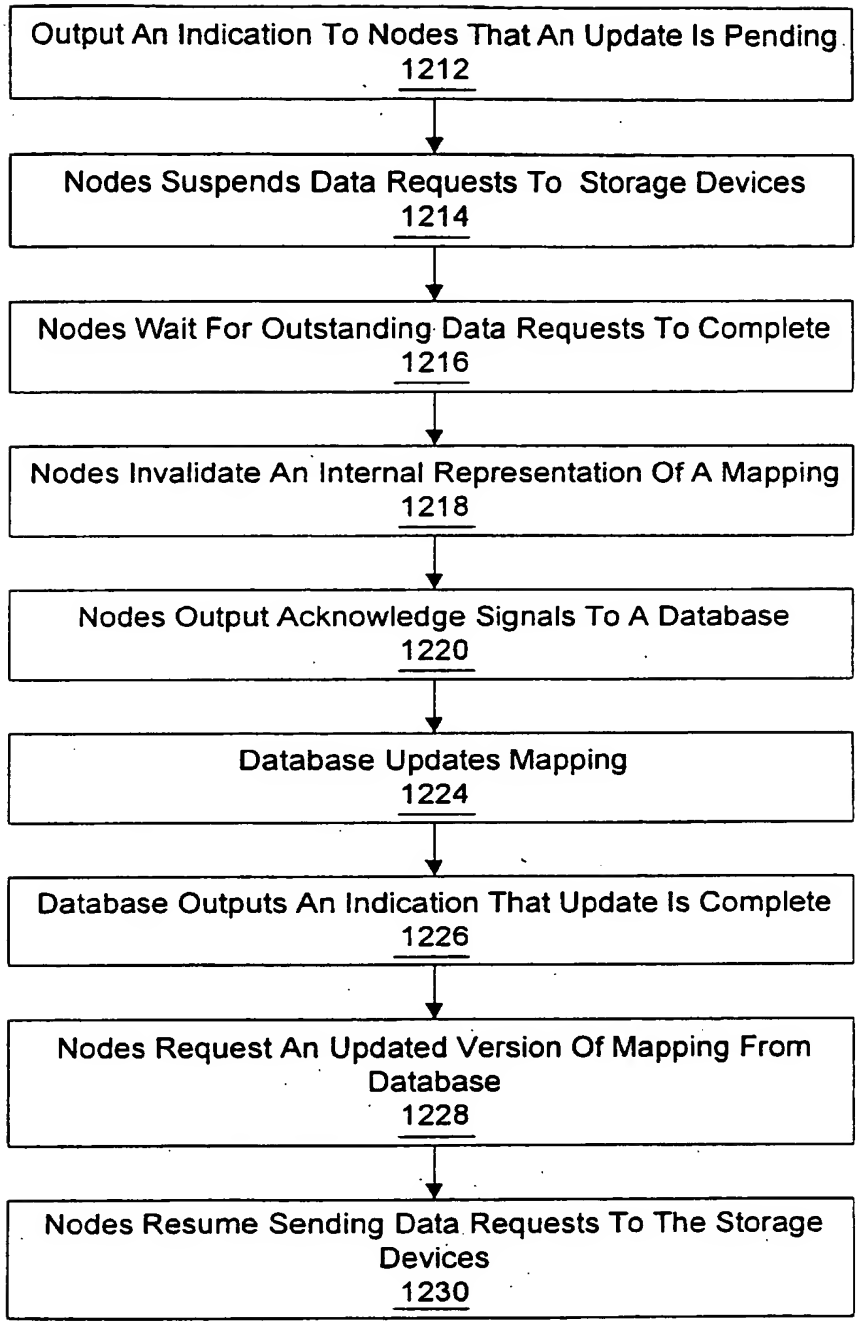
FIG. 10

【図１１】

１１１２　装置データをデータベースに記憶する
１１１４　第１のノードは装置をオープンし、データベースからの許可にアクセスする
１１１５　第１のノードは許可データを含む特殊装置ファイルを作成する
１１１６　第２のノードは装置をオープンし、データベースからの許可データにアクセスする
１１１７　第２のノードは許可データを含む特殊装置ファイルを作成する



【図１２】



１２１２　更新が保留中であるという表示をノードに出力する
１２１４　ノードは記憶装置へのデータ要求を中断する
１２１６　ノードは未処理のデータ要求が完了するのを待つ
１２１８　ノードはマッピングの内部表現を無効にする
１２２０　ノードは肯定応答信号をデータベースに出力する
１２２４　データベースはマッピングを更新する
１２２６　データベースは更新が完了したという表示を出力する
１２２８　ノードはマッピングの更新済みバージョンをデータベースから要求する
１２３０　ノードは記憶装置へのデータ要求の送信を再開する

【手続補正書】特許協力条約第３４条補正の翻訳文提出書

【提出日】平成１２年6月２６日（２０００．６．２６）

【手続補正１】

【補正対象書類名】明細書

【補正対象項目名】特許請求の範囲

【補正方法】変更

【補正内容】

【特許請求の範囲】

【請求項１】　通信リンクに結合された第１のノードと、第２のノードと、第３のノードと、

第２および第３のノードに結合され、データを記憶するように構成された記憶装置とを含み、

前記第１のノードは仮想ディスク・システムを使用して前記記憶装置にアクセスし、前記記憶装置にアクセスする一次ノードと前記記憶装置にアクセスする代替ノードを識別するマッピング・データを記憶するように構成され、

前記第１のノードは前記通信リンクを介してデータ・アクセス要求を発行するドライバを含み、前記第２および第３のノードはそれぞれ、前記記憶装置からのデータにアクセスし、前記通信リンクを介して応答を送るように構成されたマスタを含み、前記ドライバは前記マッピング・データによって識別された前記一次ノードにデータ・アクセス要求を送るように構成される、分散コンピューティング・システム。

【請求項２】　前記ドライバは前記ドライバが前記応答を受け取るまで前記データ・アクセス要求のコピーを記憶するように構成される請求項１に記載の分散コンピューティング・システム。

【請求項３】　前記ドライバは前記ドライバが前記一次ノードからの応答を受け取り損なった場合にマッピング・データによって識別された前記代替ノードに前記データ・アクセス要求を再送するようにさらに構成される請求項１または２に記載の分散コンピューティング・システム。

【請求項４】　前記一次ノードが前記第２および第３のノードのうちの一方

であり、前記代替ノードが前記第２および第３のノードのうちのもう一方である請求項１ないし３のいずれかに記載の分散コンピューティング・システム。

【請求項５】　前記マッピング・データが前記第２のノードを一次ノードとして識別し、前記マッピング・データが前記第３のノードを代替ノードとして識別する請求項１ないし４のいずれかに記載の分散コンピューティング・システム。

【請求項６】　前記第１のノードはアクティブ・ノードのリストを含むメンバシップ・データを記憶するようにさらに構成される請求項１ないし５のいずれかに記載の分散コンピューティング・システム。

【請求項７】　前記アクティブ・ノードのリストが前記一次ノードを含まない場合、前記ドライバは前記代替ノードにデータ・アクセス要求を送るように構成される請求項６に記載の分散コンピューティング・システム。

【請求項８】　前記第１、第２、および第３のノードが前記マッピング・データの同一コピーを記憶する請求項１ないし７のいずれかに記載の分散コンピューティング・システム。

【請求項９】　前記第１、第２、および第３のノードが前記マッピング・データを含むクラスタ構成データベースの同一コピーを記憶するように構成される請求項８に記載の分散コンピューティング・システム。

【請求項１０】　前記第１、第２、または第３のノードがインアクティブになったときに前記マッピング・データが更新される請求項１ないし９のいずれかに記載の分散コンピューティング・システム。

【請求項１１】　前記ドライバは、前記マッピング・データが更新されたときに前記データ・アクセス要求の送信を中断するように構成される請求項１ないし１０のいずれかに記載の分散コンピューティング・システム。

【国際調査報告】

INTERNATIONAL SEARCH REPORT

Inte tional Application No
PCT/US 99/09903

A. CLASSIFICATION OF SUBJECT MATTER
IPC 6 G06F11/14

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 6 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	EP 0 709 779 A (INTERNATIONAL BUSINESS MACHINES) 1 May 1996 (1996-05-01) column 5, line 18 -column 6, line 35	1-107
A	US 5 475 813 A (INTERNATIONAL BUSINESS MACHINES) 12 December 1995 (1995-12-12) abstract	1-3

☐ Further documents are listed in the continuation of box C.

☒ Patent family members are listed in annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
27 September 1999	04/10/1999

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Corremans, G

1

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

formation on patent family members

Inter ional Application No
PCT/US 99/09903

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 709779 A	01-05-1996	JP 8255122 A US 5668943 A	01-10-1996 16-09-1997
US 5475813 A	12-12-1995	NONE	

Form PCT/ISA/210 (patent family annex) (July 1992)

フロントページの続き

(51)Int.Cl.[7]	識別記号	F I	ﾃｰﾏｺｰﾄﾞ(参考)
Ｇ０６Ｆ 13/10	３４０	Ｇ０６Ｆ 13/10	３４０Ｂ

(31)優先権主張番号 ０９／０７６，３４６
(32)優先日 平成10年５月12日(1998. 5. 12)
(33)優先権主張国 米国（ＵＳ）
(31)優先権主張番号 ０９／０７６，２７４
(32)優先日 平成10年５月12日(1998. 5. 12)
(33)優先権主張国 米国（ＵＳ）
(81)指定国 ＥＰ(ＡＴ，ＢＥ，ＣＨ，ＣＹ，ＤＥ，ＤＫ，ＥＳ，ＦＩ，ＦＲ，ＧＢ，ＧＲ，ＩＥ，ＩＴ，ＬＵ，ＭＣ，ＮＬ，ＰＴ，ＳＥ)，ＯＡ(ＢＦ，ＢＪ，ＣＦ，ＣＧ，ＣＩ，ＣＭ，ＧＡ，ＧＮ，ＧＷ，ＭＬ，ＭＲ，ＮＥ，ＳＮ，ＴＤ，ＴＧ)，ＡＰ(ＧＨ，ＧＭ，ＫＥ，ＬＳ，ＭＷ，ＳＤ，ＳＬ，ＳＺ，ＵＧ，ＺＷ)，ＥＡ(ＡＭ，ＡＺ，ＢＹ，ＫＧ，ＫＺ，ＭＤ，ＲＵ，ＴＪ，ＴＭ)，ＡＥ，ＡＬ，ＡＭ，ＡＴ，ＡＵ，ＡＺ，ＢＡ，ＢＢ，ＢＧ，ＢＲ，ＢＹ，ＣＡ，ＣＨ，ＣＮ，ＣＵ，ＣＺ，ＤＥ，ＤＫ，ＥＥ，ＥＳ，ＦＩ，ＧＢ，ＧＤ，ＧＥ，ＧＨ，ＧＭ，ＨＲ，ＨＵ，ＩＤ，ＩＬ，ＩＮ，ＩＳ，ＪＰ，ＫＥ，ＫＧ，ＫＰ，ＫＲ，ＫＺ，ＬＣ，ＬＫ，ＬＲ，ＬＳ，ＬＴ，ＬＵ，ＬＶ，ＭＤ，ＭＧ，ＭＫ，ＭＮ，ＭＷ，ＭＸ，ＮＯ，ＮＺ，ＰＬ，ＰＴ，ＲＯ，ＲＵ，ＳＤ，ＳＥ，ＳＧ，ＳＩ，ＳＫ，ＳＬ，ＴＪ，ＴＭ，ＴＲ，ＴＴ，ＵＡ，ＵＧ，ＵＺ，ＶＮ，ＹＵ，ＺＡ，ＺＷ

(71)出願人 901 ＳＡＮ ＡＮＴＯＮＩＯ ＲＯＡＤ ＰＡＬＯ ＡＬＴＯ，ＣＡ 94303，Ｕ．Ｓ．Ａ．
(72)発明者 トラバーサット，ベルナルド・エイ
アメリカ合衆国・94109・カリフォルニア州・サン フランシスコ・カリフォルニア ストリート・2055・アパートメント 402
(72)発明者 ハーンドン，ロバート
アメリカ合衆国・80906・コロラド州・コロラド スプリングス・サウス ネバダ アベニュ 137番・1837
(72)発明者 ジェン，シャオヤン
アメリカ合衆国・94555・カリフォルニア州・フレモント・ゴルービン コモン・5454
(72)発明者 ブロック，ロバート・ジェイ
アメリカ合衆国・94043・カリフォルニア州・マウンテン ビュー・ノース レングストーフ 29番・265

Ｆターム(参考) 5B014 EA04 HC02 HC15
5B027 AA00 BB06 BB07
5B034 BB11 CC05
5B082 DA01 DE01 DE02 DE04 FA07
HA05 JA01
5B083 AA09 BB03 CD06 CE01 DD13
EE11 GG04

【要約の続き】
実行され、マッピングが変更される。次にそのノードはマッピングの内部表現を更新し、データ・アクセス要求の発行を再開する。